



ภาควิชาวิศวกรรม
 คณะครุศาสตร์อุตสาหกรรม
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ใบรับรองปริญญาโท

ชื่อหัวข้อ เครื่องมือวัดและปรับเทียบ Thermocouple
 Measuring and Calibration Temperature

ชื่อนักศึกษา 1. นายจักรพันธ์ สมุทรไชยกิจ รหัสประจำตัว 43035407
 2. นายนรินทร์ ปานทอง รหัสประจำตัว 43035415
 3. นายวินัย กระจ่างแก้ว รหัสประจำตัว 43035428
 4. นายสมพงษ์ วิจิตรโสภณ รหัสประจำตัว 43035434

หลักสูตร ครุศาสตร์อุตสาหกรรมบัณฑิต สาขาวิชา เทคโนโลยีการวัดคุมทางอุตสาหกรรม
 อาจารย์ที่ปรึกษา อาจารย์สุรพงษ์ สิริพงษ์ดี
 อาจารย์ที่ปรึกษาร่วม อาจารย์โกศล ตราชู

คณะกรรมการสอบปริญญาโท	ลายมือชื่อ
1. อาจารย์สุรพงษ์ สิริพงษ์ดี	
2. อาจารย์โกศล ตราชู	
3. อาจารย์พีระวุฒิ สุวรรณจันทร์	
4. อาจารย์อำพล ทองระอา	
5. อาจารย์ปิยะ ศุภวราสุวัฒน์	

วัน/เดือน/ปีที่สอบ วันพฤหัสบดีที่ 22 พฤศจิกายน พ.ศ. 2544 เวลา 15.00 น.

สถานที่สอบ ห้อง ค.311 คณะครุศาสตร์อุตสาหกรรม สจล.

ภาควิชารับรองแล้ว

ลงนาม.....

(ผศ.วิสุทธิ์ อธิพรธรรม)

หัวหน้าภาควิชาวิศวกรรม

วันที่ 20 เดือน สิงหาคม พ.ศ. 2544



<BT4403042>

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการศึกษาเท่านั้น ไม่สามารถนำออกจำหน่ายหรือเผยแพร่โดยไม่ได้รับอนุญาต
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เครื่องมือวัดและปรับเทียบ THERMOCOUPLE

MEASURING AND CARIBRATION THERMOCOUPLE



นายจักรพันธ์ สมุทร ไชยกิจ
นายรินทร์ ปานทอง
นายวินัย กระจ่างแก้ว
นายสมพงษ์ วิจิตรโสภณ

เลขที่.....
เลขทะเบียน 44218
วัน, เดือน, ปี 4 พ.ย. 2545

b.....
i.....

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตรบัณฑิต สาขาวิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม
ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เรื่อง เครื่องมือวัดและปรับเทียบ Thermocouple

Measuring and calibration Thermocouple

วัตถุประสงค์ของการทำปริญญานิพนธ์

1. เพื่อศึกษาในการเขียนโปรแกรมไมโครคอนโทรลเลอร์
2. เพื่อต้องการออกแบบวงจรตรวจวัดและปรับเทียบการวัดอุณหภูมิโดยใช้ Sensor เป็น Thermocouple
3. เพื่อต้องการสร้างอุปกรณ์ Calibration Type J และ K ได้
4. เพื่อใช้ทำการตรวจสอบ Thermocouple Type J และ K ได้
5. เพื่อนำไปใช้ในระบบวัดคุมทางอุตสาหกรรมได้

ประโยชน์ที่คาดว่าจะได้รับ

1. เขียนโปรแกรมและออกแบบโปรแกรมของไมโครคอนโทรลเลอร์
2. สามารถออกแบบและเขียนวงจรแผ่น PCB ได้
3. สามารถสร้างอุปกรณ์ Measuring and calibration Temperature
4. สามารถทำการใช้งานตรวจสอบ Thermocouple Type J และ K ได้
5. สามารถนำไปใช้ในระบบวัดคุมทางอุตสาหกรรมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

I

ชื่อหัวข้อ

เครื่องมือวัดและเปรียบเทียบ THERMOCOUPLE

ชื่อนักศึกษา

นายจักรพันธ์ สมุทรไชยกิจ

นายนรินทร์ ปานทอง

นายวินัย กระจ่างแก้ว

นายสมพงษ์ วิจิตรโสภณ

อาจารย์ที่ปรึกษา

อาจารย์สุรพงษ์ สิริพงศ์ดี

อาจารย์ที่ปรึกษาร่วม

อาจารย์โกศล ตราชู

หลักสูตร

ครุศาสตร์อุตสาหกรรมบัณฑิต

สาขาวิชา

เทคโนโลยีการวัดคุมทางอุตสาหกรรม

ปีการศึกษา

2544

บทคัดย่อ

ปริญญาานิพนธ์ฉบับนี้ นำเสนอเครื่องมือวัดและเปรียบเทียบ Thermocouple ประกอบไปด้วย ภาควควบคุมและภาคประมวลผล เพื่อใช้เป็นอุปกรณ์การเรียนการสอนเพื่อเพิ่มประสิทธิภาพการเรียน การสอนและสามารถ นำอุปกรณ์เครื่องมือวัดและเปรียบเทียบ Thermocouple ไปใช้ในงานอุตสาหกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

II

Thesis Title	Measuring and Caribration Thermocouple
Students	Mr.Jakkpan Samuschaiyakit Mr.Narin Panthong Mr.Winai Krajangkal Mr.Sompong Vijitsopol Mr.Surapong Siripongdee Mr.Koson Trachoo
Advisor	
Co – Advisor	
Education Level	Bachelor of Science in Industrial Education
Program in	Industrial Instrument Technology
Academic Year	2001

ABSTRACT

This thisis presents intrumentation and adjusts Thermocuople. Which consists of control system and control. Processing unit by using as studying device for improve efficient of studying and teaching. It could be token intrumentation and adjusted. Thermocouple to the industrial.

กิตติกรรมประกาศ

การจัดทำปฏิญานิพนธ์นี้สามารถสำเร็จลุล่วงด้วยดี จากสมาชิกภายในกลุ่มทุกท่าน นอกจากนี้ยังได้รับความอนุเคราะห์จาก ผศ.วิสุทธิ อธิพรธรรม อาจารย์สุรพงษ์ สิริพงศ์ดี อาจารย์โกศล ตราชู อาจารย์อำพล ทองระอา อาจารย์พีระวุฒิ สุวรรณจันทร์ และคณาจารย์ภายในภาควิชาวิศวกรรมที่ให้คำปรึกษาเกี่ยวกับโครงการและแนะนำและให้ความช่วยเหลือต่างๆ ตลอดจนให้โอกาสในการจัดทำปฏิญานิพนธ์อย่างเต็มที่ ทั้งด้านเวลา สถานที่ เครื่องมือและอุปกรณ์ต่างๆ และขอขอบพระคุณบุพการี ผู้ให้กำเนิดและให้โอกาสในทางการศึกษา ตลอดจนเพื่อนๆ ทุกคนและผู้ที่เกี่ยวข้องทุกท่านที่ให้ความช่วยเหลือและให้คำแนะนำต่างๆ และอีกทั้งยังเป็นกำลังใจในการทำปฏิญานิพนธ์จนสำเร็จลุล่วงไปด้วยดี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญรูป	VII
บทที่ 1 บทนำ	1
1.1 ความเป็นมา	1
1.2 ชัดความสามารถของโครงการ	1
1.3 เนื้อหาโดยสังเขป	2
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 กล่าวนำ	3
2.2 ไมโครคอนโทรลเลอร์ MCS – 51	3
2.2.1 โครงสร้างของ MCS – 51	3
2.2.2 การจัดขาต่างๆของ MCS – 51	4
2.2.3 โครงสร้างของพอร์ตอินพุตและเอาต์พุต	6
2.2.4 โครงสร้างของหน่วยความจำ	7
2.2.5 หน่วยความจำทั่วไป	7
2.2.6 รีจิสเตอร์ฟังก์ชันพิเศษ	9
2.2.7 หน่วยความจำภายนอก	12
2.2.8 Reset Operation	14
2.3 เทอร์โมคัปเปิล	15
2.3.1 ประวัติการค้นพบเทอร์โมคัปเปิล	15
2.3.2 กฎการเกิดแรงเคลื่อนไฟฟ้าจากความร้อน	15
2.3.3 ชนิดของเทอร์โมคัปเปิลแบบมาตรฐาน	15
2.3.4 เทอร์โมคัปเปิลและส่วนประกอบ	16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
2.3.5 อาร์ทีดี	16
บทที่ 3 การออกแบบ การสร้าง และการทำงาน	18
3.1 ฮาร์ดแวร์	18
3.1.1 การออกแบบวงจร A/D 8 Bit	19
3.1.2 การออกแบบวงจร D/A 8 Bit	20
3.1.3 วงจรแหล่งจ่ายตัวสร้างอนุกรม	20
3.1.4 วงจรแหล่งจ่ายตัว Calibration	21
3.1.5 การออกแบบกล่องตัวสร้างอนุกรม	22
3.2 ซอฟต์แวร์	23
3.2.1 ส่วนของกล่องตัวสร้างอนุกรม	23
3.2.2 ส่วนของ Calibration	24
บทที่ 4 การทดลองและผลการทดลอง	26
4.1 การทดลองและการทดสอบ	26
4.1.1 อุปกรณ์ที่ใช้ในการทดลอง	26
4.1.2 ลำดับขั้นตอนการทดลอง	26
บทที่ 5 บทสรุป ปัญหา แนวทางการแก้ไขและการพัฒนา	30
5.1 บทสรุป	30
5.2 ปัญหาและแนวทางแก้ไข	31
5.3 แนวทางในการพัฒนา	31
ภาคผนวก ก รูปต้นแบบของเครื่องวัดและเปรียบเทียบ Thermocouple	32
ภาคผนวก ข วงจรการทำงาน	35
ภาคผนวก ค คู่มือการใช้งานเครื่องวัดและเปรียบเทียบ Thermocouple	38
ภาคผนวก ง โปรแกรมการทำงาน	42
บรรณานุกรม	115
ประวัติผู้แต่ง	116

สารบัญตาราง

ตาราง

หน้า

ตารางที่ 4.1 ผลการทดลอง A/D 8Bit

27



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป	หน้า
รูปที่ 2.1 โครงสร้างภายใน MCS – 51	4
รูปที่ 2.2 ตำแหน่งขาต่างๆของ MCS – 51	5
รูปที่ 2.3 พอร์ตอินพุตและเอาต์พุต	6
รูปที่ 2.4 บัสอินพุต MCS – 51	6
รูปที่ 2.5 โครงสร้างหน่วยความจำ	8
รูปที่ 2.6 การใช้ MCS – 51 เปิดปิดหลอดไฟ	11
รูปที่ 2.7 แผนผังกลุ่มสัญญาณที่ใช้อ่านข้อมูล	13
รูปที่ 2.8 แผนผังเวลาการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอก	14
รูปที่ 3.1 ผังแสดงการทำงานของเครื่องมือวัดและปรับเทียบ Thermocouple ในส่วนของตัวสร้างอุณหภูมิ	18
รูปที่ 3.2 ผังแสดงการทำงานของเครื่องมือวัดและปรับเทียบ Thermocouple ในส่วนของ Calibration	19
รูปที่ 3.3 ADC 8 Bit	19
รูปที่ 3.4 การใช้งาน D/A 8 Bit	20
รูปที่ 3.5 วงจรแหล่งจ่ายตัวสร้างอุณหภูมิ	21
รูปที่ 3.6 วงจรแหล่งจ่ายตัว Calibration	21
รูปที่ 3.7 รูปด้านล่างของกล่องสร้างอุณหภูมิ	22
รูปที่ 3.8 ส่วนตัวกล่องตัวสร้างอุณหภูมิ	23
รูปที่ 3.9 แผนผังการทำงานของตัวสร้างอุณหภูมิ	24
รูปที่ 3.10 แผนผังการทำงานของตัว Calibration	25
รูปที่ 4.1 รูปการทำงานของ Tempcontrol	28
รูปที่ 4.2 รูปการทำงานของเครื่อง Calibration	28
รูปที่ 4.3 ลักษณะของการ Set ของเครื่อง Calibration	29
รูปที่ 4.4 ลักษณะของ Display ของเครื่อง Calibration	29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป	หน้า
รูปที่ ก.1 ส่วนของห้อง Heater	33
รูปที่ ก.2 เครื่อง T.C. Calibration	33
รูปที่ ก.3 เครื่อง Temp Control	34
รูปที่ ก.4 การต่อใช้งานเพื่อทำการ Calibration Thermocouple	34
รูปที่ ข.1 แผนผังการทำงานของเครื่องมือวัดและปรับเทียบ Thermocouple	36
รูปที่ ข.2 แผนผังการทำงานของส่วนตัว Calibration	37
รูปที่ ค.1 ส่วนประกอบด้านหน้า	39
รูปที่ ค.2 ส่วนประกอบด้านหลัง	39



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์

ในปัจจุบันอุปกรณ์ที่ใช้เป็นตัวมาตรฐานที่นำมาเปรียบเทียบกับ Thermocouple ที่ใช้ในงานอุตสาหกรรมเป็นเครื่องมือที่มีราคาแพง และบางครั้งต้องนำเข้ามาจากต่างประเทศ ซึ่งถ้านำมาจากต่างประเทศก็จะมีราคาแพงและเสียเวลาในการขนส่งสูง และการนำเอาอุปกรณ์ต่างประเทศมานั้นก็จะมีปัญหาคือเสียเวลา ในบางครั้งถ้าอุปกรณ์มีปัญหาที่ต้องเสียเวลาส่งไปซ่อมถึงต่างประเทศด้วยทำให้เป็นการเสียเวลาและเสียค่าใช้จ่ายมากขึ้นด้วย ดังนั้น เราจึงเกิดแนวคิดริเริ่มประดิษฐ์อุปกรณ์ Measuring Calibration Temperature Type K และ J และ Heater อุปกรณ์ที่ใช้ในการสร้างเครื่อง Calibration และ Heater นั้น ประกอบด้วยภาค Control และ ภาค Power ซึ่งภาค Control เป็นส่วนที่มีความสำคัญมากเพื่อให้เกิดค่า Error น้อยที่สุดจึงเลือกใช้ Microcontroller โครงการนี้จะเน้นประโยชน์กับระบบการวัดคุมทางอุตสาหกรรม และการเรียนการสอนในวิชาเครื่องมือวัดและควบคุมทางอุตสาหกรรมต่าง ๆ ในระดับประกาศนียบัตรวิชาชีพชั้นสูงและระดับปริญญาตรี ซึ่งจะเป็นอุปกรณ์ Test Thermocouple Type K และ J และยังสามารถสร้างอุณหภูมิให้กับอุปกรณ์ Sensor ในงานวัดและควบคุมได้อีกด้วย

1.2 ขีดความสามารถของโครงการ

โครงการ เครื่องมือวัดและปรับเทียบ Thermocouple มีขีดความสามารถดังนี้

- 1) ค่าความคลาดเคลื่อนของเครื่องวัดและปรับเทียบ Thermocouple $\pm 5\%$
- 2) มีความละเอียดในการแสดงผลเท่ากับทศนิยม 3 ตำแหน่ง
- 3) สามารถตรวจสอบ Thermocouple Type K และ J ได้
- 4) แสดงผลเป็น LED 1 บรรทัด
- 5) สามารถแสดงผลค่า Standard Table ของ Thermocouple Type K และ J ได้
- 6) สามารถสร้างอุณหภูมิสูงสุดประมาณ 250°C
- 7) สามารถตั้งอุณหภูมิได้ตามต้องการ

1.3 เนื้อหาโดยสังเขป

เนื้อหาในปฏิญญานิพนธ์ฉบับนี้ แบ่งออกเป็นบทต่าง ๆ เพื่อความสะดวกต่อการศึกษาและทำความเข้าใจ ในแต่ละบทจะประกอบด้วยเนื้อหาสำคัญดังนี้

บทที่ 2 ทฤษฎีและหลักการ เป็นเนื้อหาที่เกี่ยวข้องกับทฤษฎี และหลักการของส่วนประกอบของโครงสร้างของโครงการ ประกอบไปด้วยไมโครคอนโทรลเลอร์ตระกูล MCS-51 ตัวเทอร์โมคัปเปิล และส่วนประกอบ อาร์ทีดี

บทที่ 3 การออกแบบและการสร้าง กล่าวถึงขั้นตอนการออกแบบฮาร์ดแวร์และซอฟต์แวร์ ทั้งวงจรการทำงานต่างๆ

บทที่ 4 การทดลองและผลการทดลอง เป็นส่วนที่ทำการทดสอบการทำงานของเครื่องมือวัด และเปรียบเทียบ Thermocouple

บทที่ 5 บทสรุป ปัญหา แนวทางการแก้ไขและพัฒนา กล่าวสรุปผลการทำโครงการ ตลอดจนปัญหาที่เกิดขึ้นระหว่างการดำเนินการ ตลอดจนแนวทางแก้ไขปัญหาที่เกิดขึ้น รวมทั้งข้อเสนอแนะ เพื่อเป็นแนวทางในการพัฒนาของเครื่องมือวัดและเปรียบเทียบ Thermocouple ให้สามารถนำไปใช้ในงานอุตสาหกรรมต่อไปได้

ภาคผนวก ก รูปต้นแบบของเครื่องมือวัดและเปรียบเทียบ Thermocouple

ภาคผนวก ข วงจรการทำงาน

ภาคผนวก ค คู่มือการใช้งาน

ภาคผนวก ง โปรแกรมการทำงาน

บรรณานุกรม

ประวัติผู้แต่ง

บทที่ 2

ทฤษฎีและหลักการ

2.1 กล่าวนำ

ประกอบด้วยเนื้อหาในทางทฤษฎี แนวความคิด ที่นำมาประกอบโครงสร้างโครงการ โดยประกอบด้วยไมโครคอนโทรลเลอร์ตระกูล MCS-51 เพื่อที่จะเป็นตัวประมวลผลของ เครื่องมือวัดและปรับเทียบ Thermocouple และ เพื่อที่จะเป็นตัวประมวลผลของเครื่อง Heater และวงจรควบคุมของเครื่อง Heater

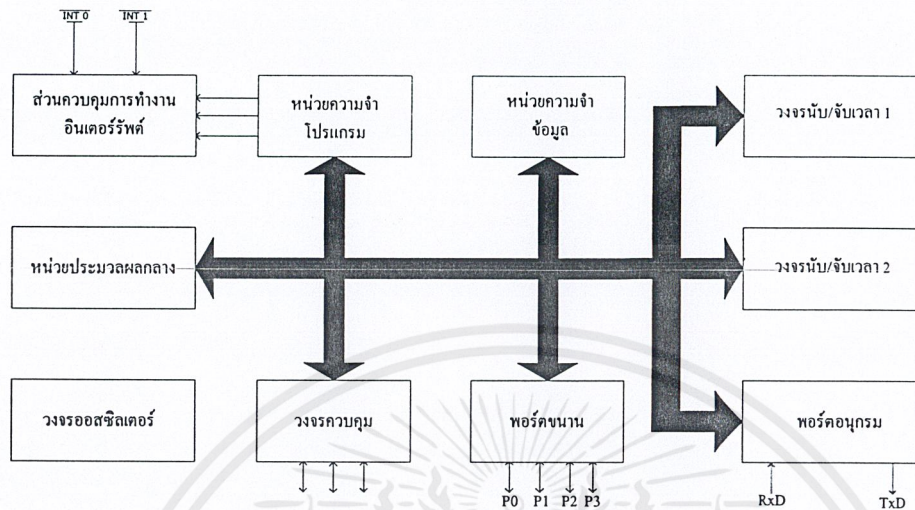
2.2 ไมโครคอนโทรลเลอร์ MCS-51

2.2.1 โครงสร้างของ MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีด้วยกันหลายเบอร์ขึ้นกับโครงสร้างภายในของมัน บางเบอร์จะมีหน่วยความจำภายในแบบ ROM บางเบอร์เป็นแบบ EPROM บางเบอร์มี RAM ภายใน 128 ไบต์ บางเบอร์มี 256 ไบต์ เป็นต้น คุณสมบัติที่สำคัญของ MCS-51 มีดังนี้

- 1) มีหน่วยความจำ ROM 4K bytes
- 2) มีหน่วยความจำ RAM 128 bytes
- 3) มีพอร์ต I/O ขนาด 8 บิต 4 พอร์ต
- 4) มี Timer 16 บิต 2 ตัว
- 5) สามารถอินเทอร์รัพท์ได้ 5 แหล่ง
- 6) มีวงจรออสซิลเลเตอร์และวงจรมานาฬิกาบนชิพ
- 7) มีพอร์ตอนุกรมที่สามารถรับส่งข้อมูลแบบ Full Duplex ความเร็วสูง
- 8) อ้างหน่วยความจำโปรแกรมภายนอกได้ 64K
- 9) อ้างหน่วยความจำข้อมูลภายนอกได้ 64K
- 10) สามารถประมวลผลทีละบิตได้
- 11) สามารถอ้างหน่วยความจำแบบบิตได้ 210 ตำแหน่ง

12) หนึ่งวัฏจักรคำสั่งกินเวลาประมาณ 1 ไมโครวินาที ขณะทำงานด้วย Clock 12 MHz



รูปที่ 2.1 โครงสร้างภายในของ MCS- 51

2.2.2 การจัดหาต่างๆ ของ MCS-51

ไอซี ไมโครคอนโทรลเลอร์ 8051 โครงสร้าง IC เป็นแบบ DIP มีขาทั้งหมด 40 ขาโดยขาต่างๆ จะใช้เป็นขาพอร์ตอินพุตและเอาต์พุต ขาสัญญาณควบคุม ขาดำเนินหน่วยความจำ และขาข้อมูล ดังรูปที่ 2.2

1) ความหมายของขาต่างๆมีดังนี้

1.1) พอร์ต 0 (Port 0) พอร์ต 0 ได้แก่ขาที่ 32-39 ของ MCS-51 สามารถใช้เป็นอินพุตเอาต์พุต ได้นอกจากนี้ในการติดต่อกับหน่วยความจำภายนอกยังใช้เป็นขา Address Bus อีกด้วย

1.2) พอร์ต 1 (Port 1) พอร์ต 1 ได้แก่ขาที่ 1-8 เป็นพอร์ต 8 บิต สามารถอ้างทีละบิตได้ คือ P1.0,P1.1,P0.17

1.3) พอร์ต 2 (Port 2) พอร์ต 2 ได้แก่ขาที่ 21-28 จะใช้งาน 2 หน้าที คือใช้เป็นพอร์ต 8 บิต กับใช้ขาแอดเดรส 8 บิต ในการอ้างหน่วยความจำภายนอก

1.4) พอร์ต 3 (Port 3) พอร์ต 3 ได้แก่ขาที่ 10-17 จะใช้งานสองหน้าที่ก็คือ เป็นอินพุตและ เอาต์พุต และใช้เป็นขาควบคุมต่างๆ

1.5) พอร์ต 3 (Port) พอร์ต 3 ได้แก่ขาที่ 10-17 จะใช้งานสองหน้าที่คือ เป็นอินพุตและเอาต์พุต และใช้เป็นขาควบคุมต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) PSEN (Program Store Enable)

ขา ESPN เป็นขาที่ส่งสัญญาณออกคือขาที่ 29 ขานี้จะแอกทีฟเมื่อ MCS-51 ต้องการอ่าน Code โปรแกรมภายนอก โดยปกติถ้าหน่วยความจำภายนอกเป็น EPROM ขา PSEN จะต่อกับขา Output Enable (OE) ของ EPROM

3) ALE (Address Latch Enable)

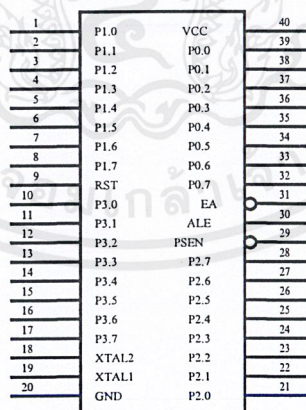
เนื่องจากพอร์ต 0 สามารถใช้เป็นขาอ้างอิงตำแหน่ง และขาข้อมูล จะมีขา ALE ได้แก่ขา 30 ขานี้จะใช้ Multiplex สัญญาณ Address Bus ของ Port 0 ในการใช้งานระบบ MCS-51 นั้น จะต้องมียูปรกรณ์มาต่อ

4) EA (External Access)

ขา EA ได้แก่ขาที่ 31 ถ้าขานั้นเป็นลอจิก “1” จะใช้กับเบอร์ 8051/8052 เพื่อบอกว่าให้อ่านโปรแกรมจากหน่วยความจำโปรแกรมภายใน แต่ถ้าเป็นลอจิก “0” จะบอกให้ว่า MCS-51 ทำโปรแกรม โดยอ่านจากหน่วยความจำโปรแกรมภายนอก (ถ้าขา EA เป็น “0” ขา PSEN จะแอกทีฟ) ถ้าหากเป็นเบอร์ 8051/8052 ซึ่งมีหน่วยความจำโปรแกรมภายในและให้ขา EA เป็น “0” ซึ่งจะ Disabled ROM ภายในและจะอ่านโปรแกรมจาก EPROM ภายนอกแทน

5) RST (RESET)

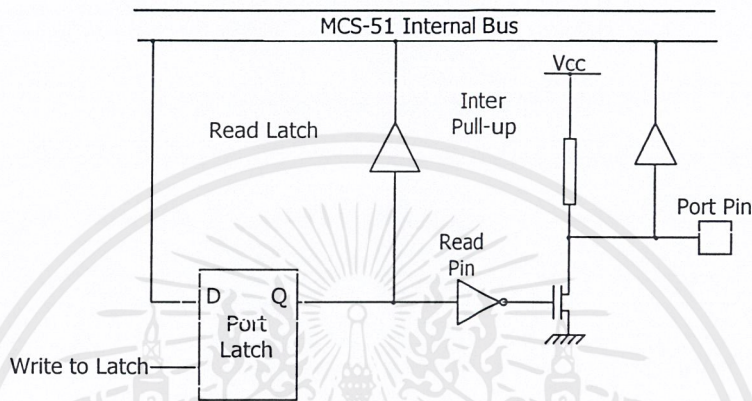
ขา RST ได้แก่ขา 9 จะใช้ในการรีเซ็ต MCS-51 โดยจะให้ขานี้เป็นลอจิก “1” อย่างน้อย 2 Machine Cycles จึงจะรีเซ็ตระบบได้



รูปที่ 2.2 ตำแหน่งขาต่างๆ ของ MCS-51

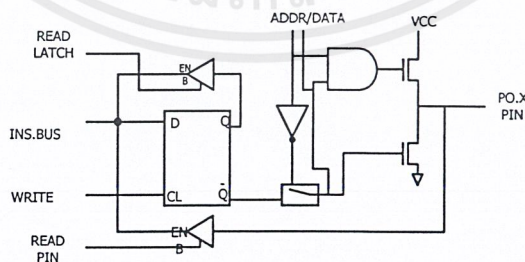
2.2.3 โครงสร้างของพอร์ทอินพุตเอาต์พุต (I/O Port Structure)

ขาของพอร์ทแสดงโครงสร้างภายในได้ ดังรูปที่ 2.3 โดยจะมีโครงสร้างเป็น Field-effect Transistor ต่ออยู่กับขาภายนอกและมีความต้านทานต่อ Pull-up อยู่สำหรับพอร์ท 1 2 3 แต่ถ้าพอร์ท 0 และจะไม่มีตัวต้านทาน Pull-up ภายใน เพราะว่าต้องใช้เป็นขา Address Bus และ Data Bus



รูปที่ 2.3 พอร์ทอินพุตและเอาต์พุต

พอร์ทนี้สามารถให้เป็นสัญญาณอินพุตกับอุปกรณ์ภายนอกได้ในการอ่านข้อมูลจากพอร์ทจะอ่านได้สองแบบ คือ Read Latch และ Read Pin โดย Read Latch หมายถึงการอ่านข้อมูลที่ถูกลatch เอาไว้เข้าสู่ ภายในของ MCS-51 เช่นการทำคำสั่ง CPL P1.5 แต่ถ้าเป็นการ Read Pin จะเป็นการใช้พอร์ทเป็นอินพุต โดยจะอ่านค่าจากขาไอซีเข้าสู่บัสภายในโดยการอ่านแบบ Read Latch และ Read Pin จะมีสัญญาณควบคุมที่บัสเฟอรรังรูปที่ 2.4



รูปที่ 2.4 บัสอินพุต MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 โครงสร้างหน่วยความจำ

หน่วยความจำสำหรับ MCS-51 จะมี 2 ชนิด คือ ใช้หน่วยความจำที่ใช้โปรแกรม (ROM) กับใช้หน่วยความจำที่ใช้เก็บข้อมูลในการประมวลผล ใน 8051 จะมีหน่วยความจำภายในตั้งแต่ตำแหน่ง 00H ถึง FFH และสามารถอ้างหน่วยความจำโปรแกรมภายนอกได้ 64K ตำแหน่ง สำหรับหน่วยความจำข้อมูลภายในนั้นจะแบ่งออกได้ดังนี้

- 1) ชุดรีจิสเตอร์ 4 ชุดแต่ละชุดเรียกว่ารีจิสเตอร์แบงก์ ที่ตำแหน่ง 00H ถึง 1FH โดยแต่ละชุดประกอบด้วยรีจิสเตอร์ R0 ถึง R7
- 2) หน่วยความจำที่สามารถเข้าถึงข้อมูลระดับบิตได้ ตำแหน่ง 20H ถึง 2FH
- 3) หน่วยความจำใช้งานทั่วไปตำแหน่ง 30H ถึง FFH
- 4) รีจิสเตอร์ฟังก์ชันพิเศษ ตำแหน่ง 80H ถึง FFH

2.2.5 หน่วยความจำใช้งานทั่วไป

จะเห็นได้ว่า 8051 จะมีหน่วยความจำ RAM สำหรับใช้งานทั่วไปจำนวน 80 ไบต์ ตั้งแต่ตำแหน่ง 30H ถึง 7FH ตัวอย่างเช่น ถ้าต้องการอ่านข้อมูลที่อยู่ในตำแหน่ง 5FH มาเก็บในรีจิสเตอร์ A ซึ่งสามารถเขียนคำสั่งได้เป็น

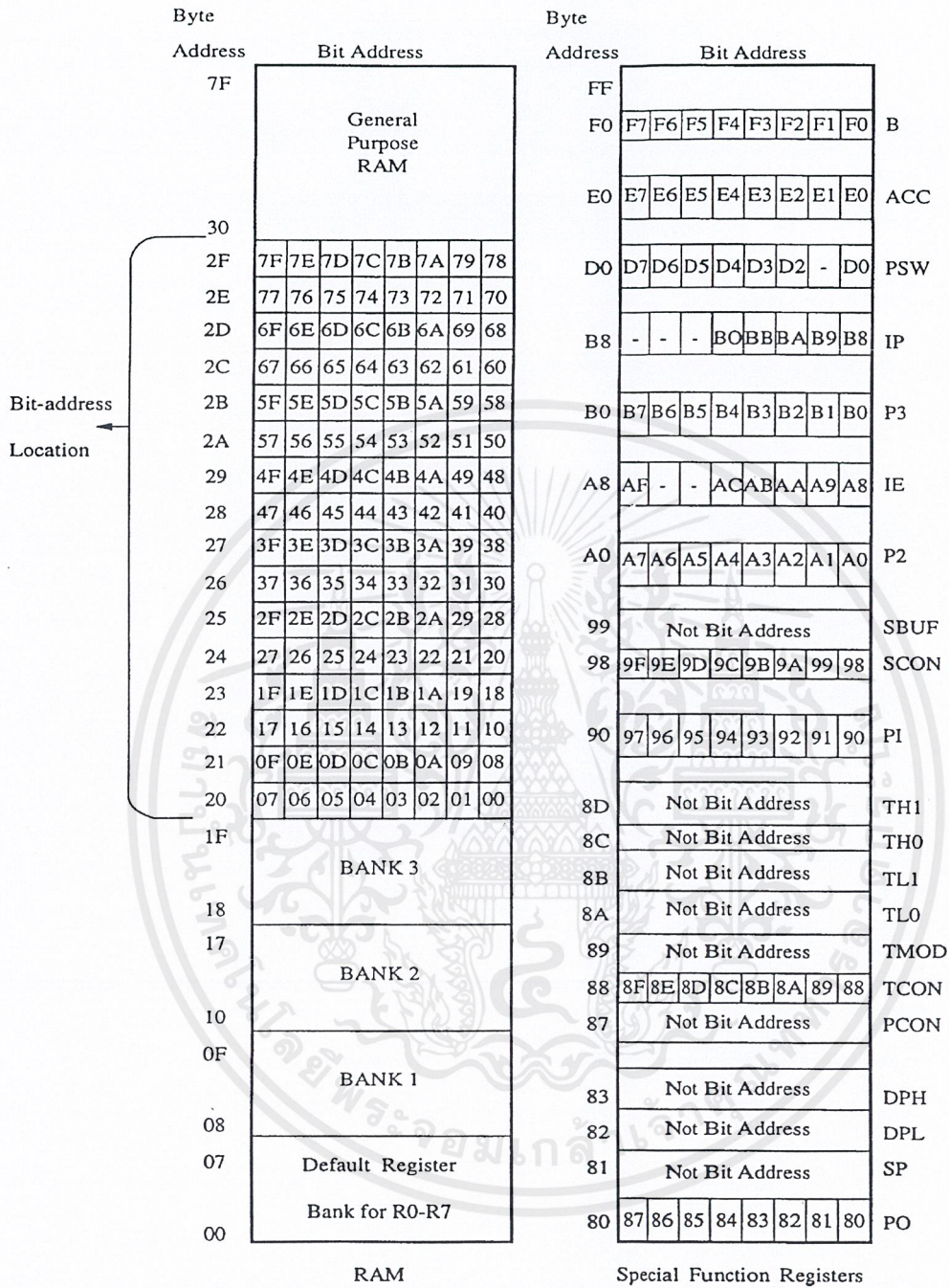
```
MOV A,5FH
```

การย้ายข้อมูลนี้เป็นการย้ายข้อมูลจากตำแหน่งที่เก็บโดยตรง (ตำแหน่ง 5FH) เรียกว่าการอ้างตำแหน่งแบบ Direct Addressing Mode นอกจากนี้ยังสามารถอ่านข้อมูลโดยใช้รีจิสเตอร์ R0 หรือ R1 การชี้ตำแหน่งได้ เรียกว่าการอ้างตำแหน่งแบบ Indirect Addressing Mode ตัวอย่างเช่น

```
MOV R0,#5FH
```

```
MOV A,@R0
```

การเขียนโปรแกรมด้านบน หมายความว่า เก็บค่า 5FH ไว้ใน R0 จากนั้นอ่านค่าที่ R0 ซึ่งอยู่คือตำแหน่ง 5FH มาเก็บไว้ในรีจิสเตอร์ A ถ้าในตำแหน่ง 5FH มี 80 อยู่ ค่า 80 จะถูกเก็บใน A



รูปที่ 2.5 โครงสร้างหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) Bit-addressable RAM

ในหน่วยความจำที่สามารถอ้างข้อมูลได้ในระดับบิตนั้น ได้ตั้งแต่ตำแหน่ง 20H ถึง 2FH รวม 16 ไบต์ โดยสามารถ SET, CLEAR, AND, OR ทางลอจิกได้ และจำนวนบิตที่ใช้งานได้ทั้งหมดมีจำนวน 128 บิต (8บิตX16ไบต์) ถ้าต้องการเซตบิตตำแหน่งที่ 67H สามารถเขียนคำสั่งได้ดังนี้

SETB 67H

2) Register Banks

หน่วยความจำข้อมูลภายในเป็นชุดของรีจิสเตอร์ ซึ่งมีทั้งหมด 32 ตำแหน่งโดยจะมี 4 ชุด แต่ละชุดมีรีจิสเตอร์ 8 ตัว คือ R0 ถึง R7 โดยชุดแรกจะอยู่ในตำแหน่ง 00H-07H ถ้าหากอ่านค่าจากตำแหน่ง 05H มาเก็บไว้ในรีจิสเตอร์ A จะเขียนโปรแกรมได้ดังนี้

MOV A,R5

2.2.5 รีจิสเตอร์ฟังก์ชันพิเศษ (Special Function Register)

ใน MCS-51 รีจิสเตอร์จะใช้หน่วยความจำ RAM ภายในชิพ โดยส่วนหนึ่งเป็น รีจิสเตอร์พิเศษ (Special- Function Register : SFR) ซึ่งมีทั้งหมด 21 ตัว โดยเฉพาะตัวรีจิสเตอร์พิเศษต่างๆจะเริ่มที่จากหน่วยความจำตั้งแต่ 80H ถึง FFH ซึ่งทั้งหมด 128 ตำแหน่ง ซึ่งแต่ละตัวจะเป็นรีจิสเตอร์ฟังก์ชันพิเศษเพียง 21 ตำแหน่ง แต่ถ้าเป็น 8032/8051 จะใช้ 26 ตำแหน่งหรือมี SFR 26 ตัว

1) Program Status Word

รีจิสเตอร์ตัวนี้เรียกละย่อว่า PSW จะอยู่ในตำแหน่ง D0H ซึ่งสามารถเข้าถึงข้อมูลระดับบิตได้โดยรีจิสเตอร์นี้จะเป็นตัวบอกสถานะต่างๆ ของไมโครคอนโทรลเลอร์ ความหมายของแต่ละบิต

1.1) แพลกตัวทอด Carry Flag (CF) บิตนี้เป็นบิตที่ 7 ของ PSW บิตนี้มีความสำคัญหากมีการกระทำทางวิทยาศาสตร์โดยบิตนี้จะ Set เมื่อเกิดการทดของบิตที่ 7 ขณะทำการบวกเลข หรือ Set เมื่อเกิดการยืมของบิตที่ 7 เมื่อเกิดการลบเลข

1.2) แพลกตัวช่วยทด Auxiliary Carry Flag เมื่อถูกการบวกแบบ Binary-Code-Decimal (BCD) บิต Auxiliary Carry Flag (AC) หรือบิตตัวช่วยทดจะถูก Set เมื่อมีการทดจากบิตที่ 3 ไปบิตที่ 4 หรือถ้าใน Lower Nibble มีค่าระหว่าง 0AH-0FH เนื่องจากรหัส BCD นี้มีค่าได้มากที่สุดแค่ 9 ถ้าหากมีการหารบวกเลขในแบบของ BCD ซึ่งจะต้องตามด้วยคำสั่ง DAA (Decimal Adjust Accumulator) เพื่อทำการปรับค่าที่มีค่าเกิน 9 โดยบวกเลข 6 เข้าไป จะทำให้เป็นรหัส BCD ที่แทนเลขฐานสิบได้

1.3) แพลกศูนย์ Flag 0 เป็น Flag ที่ผู้ใช้สามารถใช้งานได้ทั่วไป

1.4) บิตเลือกรีจิสเตอร์แบงก์ (Register Bank Select Bits) ที่ทราบมาแล้วว่าใน MCS-51 จะมีชุดรีจิสเตอร์อยู่ 4 ชุด ถ้าเลือกให้ชุดใดแอกทีฟจะกำหนดได้ในบิต RS1 และ RS2 ของ PSW และจะ

Clear ตัวเองเมื่อระบบถูกรีเซต ถ้าหากต้องการติดต่อกับรีจิสเตอร์ Bank3 โดยย้ายข้อมูลจาก R7 (ตำแหน่ง 1FH) มาเก็บใน Accumulator

1.5) แฟล็กโอเวอร์โฟลว์ Overflow Flag แฟล็ก OV จะถูก Set หลังจากการกระทำทางคณิตศาสตร์แล้วเกิด Overflow คือจำนวนที่เกิดจากการบวกหรือการลบ มีค่าเกินกว่าที่จำนวนไบต์จะเป็นไปได้คือ มากกว่า +128 หรือน้อยกว่า -128

1.6) บิตพาริตี (Parity Bit) พาริตีบิต (P) ซึ่งเป็นบิตที่บอกถึงค่าพาริตีบิตของรีจิสเตอร์ Accumulator ซึ่งอาจเป็นตัวตรวจสอบความถูกต้องของข้อมูลได้ โดยจะเซตหรือเคลียร์ ขึ้นอยู่กับผลที่เกิดขึ้นกับ Accumulator

2) รีจิสเตอร์ B (B Register)

รีจิสเตอร์ B จะอยู่ตำแหน่ง FOH ของหน่วยความจำข้อมูลภายใน เป็นรีจิสเตอร์ที่สามารถใช้งานทั่วไปได้ โดยทั่วไปรีจิสเตอร์นี้จะใช้คูณหรือหารกับรีจิสเตอร์ Accumulator

3) ตัวชี้สแตค (Stack Pointer)

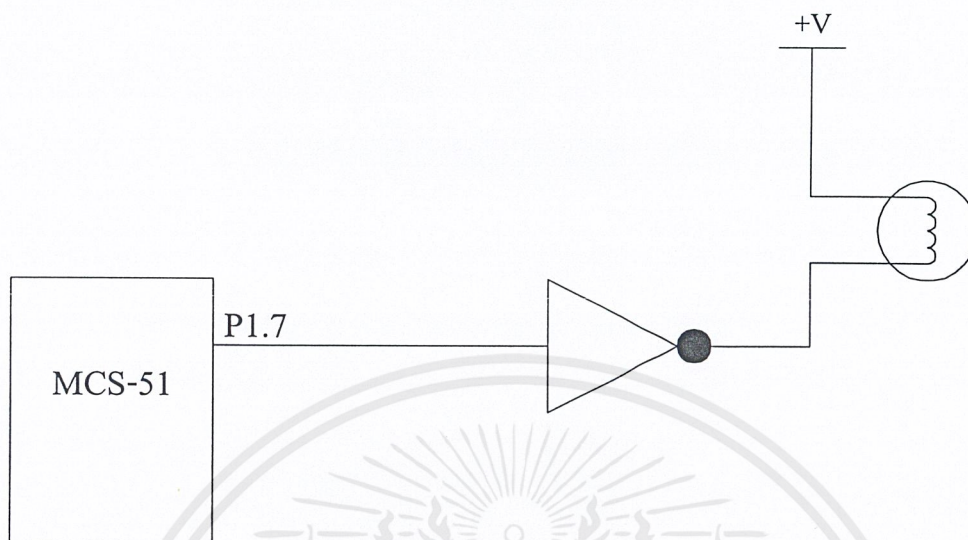
Stack Pointer (SP) เป็นรีจิสเตอร์ขนาด 8 บิต อยู่ในตำแหน่ง 81H การเขียนค่าเข้าไปในตำแหน่งที่ SP ซ้ำอยู่ตำแหน่ง เรียกว่า “Pushing” สำหรับการอ่านค่า SP จะเพิ่มขึ้นหนึ่งก่อนที่จะเขียนข้อมูลลงไป และจะลดลงหนึ่งเมื่ออ่านข้อมูลมาแล้ว

4) รีจิสเตอร์ Data Pointer (DPTR)

รีจิสเตอร์ใช้สำหรับชี้ตำแหน่งรหัสโปรแกรมหรือข้อมูลในหน่วยความจำต่างๆ โดยเป็นรีจิสเตอร์ขนาด 16 บิต ซึ่งประกอบด้วยรีจิสเตอร์ 2 ตัว คือ DPL ตำแหน่งที่ 82H โดยจะเก็บเป็น 8 บิตค่า และ DPH ตำแหน่งที่ 83H โดยจะเก็บค่า 8 บิตสูง รีจิสเตอร์ทั้งสองตัวนี้จะรวมกันกลายเป็นรีจิสเตอร์ 16 บิต ถ้าหากต้องการ เก็บค่า 55H ไปยังตำแหน่งของหน่วยความจำข้อมูลภายนอกตำแหน่งที่ 1000H

5) รีจิสเตอร์พอร์ต (Port Register)

ใน MCS-51 ค่าของพอร์ตจะหมายถึงค่าของความจำด้วย หากต้องการส่งข้อมูล 1 ออกไปที่พอร์ต ก็เพียงแต่เขียนข้อมูลไปที่หน่วยความจำตำแหน่งที่พอร์ตนั้นอยู่ และถ้าหากต้องการจะอ่านข้อมูลจากพอร์ต ก็เพียงอ่านค่าจากตำแหน่งที่หน่วยความจำที่พอร์ตนั้นอยู่ ใน MCS-51 พอร์ต 0 จะอยู่ที่ตำแหน่ง 80H พอร์ต 1 จะอยู่ที่ตำแหน่ง 90H พอร์ต 2 จะอยู่ที่ตำแหน่ง A0H และพอร์ต 3 จะอยู่ที่ตำแหน่ง B0H พอร์ต 0 2 และ 3 โดยทั่วไปแล้วจะไม่ใช่ถ้าหากมีการติดต่อกับหน่วยความจำภายนอกหรือใช้เป็นพอร์ตพิเศษ โดยปกติแล้วจะใช้พอร์ต 1 ในการติดต่อกับอุปกรณ์ภายนอก พอร์ตทุกพอร์ตสามารถอ้างข้อมูลระดับบิตได้ ดังตัวอย่างเช่น ถ้าพอร์ต 1 บิต 7 ต่อกับหลอดไฟไว้ด้วยรูป โดยมีเกดแบบ OC ช่วยขับกระแสการปิดเปิด หลอดไฟทำได้โดยการ Set หรือ Clear บิต 7 ของพอร์ต 1 นี้



รูปที่ 2.6 การใช้ MCS-51 เปิดปิดหลอดไฟ

6) รีจิสเตอร์เวลา (Timer Register)

ใน MCS-51 เบอร์ 8051 จะมีรีจิสเตอร์ที่ใช้นับและจับเวลาขนาด 16 บิต 2 ตัวคือ Timer 0 อยู่ที่ตำแหน่ง 8AH และ 8CH โดยตำแหน่ง TL0 ซึ่งจะเป็น 8 ไบต์ต่ำ และ 8CH หมายถึง 8 ไบต์สูง TH0 รีจิสเตอร์อีกตัวคือ Timer 1 โดยแบ่งเป็น TL1 อยู่ที่ตำแหน่ง 8BH เป็นไบต์ต่ำ และ TH1 อยู่ที่ตำแหน่ง 8DH จะเป็นไบต์สูง และการใช้ Timer จะต้องกำหนดการทำงานในค่ารีจิสเตอร์ TMOD (Timer/Counter Mode Control Register) ซึ่งอยู่ที่ตำแหน่ง 88H เสียก่อน

7) รีจิสเตอร์อนุกรม (Serial Port Register)

MCS-51 จะมีพอร์ตสื่อสารอนุกรม (Serial Register) อยู่ภายในชิพ ซึ่งสามารถจะรับหรือส่งข้อมูลได้โดยติดต่อผ่านรีจิสเตอร์ SBUF (Serial Data Buffer) ซึ่งอยู่ที่ตำแหน่ง 99H โดยถ้าต้องการส่งข้อมูลแบบอนุกรมให้เขียนรีจิสเตอร์นี้ ตัว Serial Port สามารถโปรแกรมให้ทำงานได้ 4 โหมด โดยโปรแกรมผ่าน รีจิสเตอร์ SCON (Serial Port Control Register) ตำแหน่ง 98H

8) รีจิสเตอร์อินเทอร์รัพท์ (Interrupt Port Register)

MCS-51 สามารถ Interrupt ได้ 5 ตำแหน่ง โดยมี 2-Priority ตัว Interrupt นี้จะถูก Disable หลังจากที่ระบบซึ่งถูกรีเซต และจะ Enabled หลังจากที่เขียนข้อมูลไปที่รีจิสเตอร์ IE หรือตำแหน่ง A8H

ลำดับความสำคัญสามารถเซตได้ที่รีจิสเตอร์ IP หรือตำแหน่ง B8H

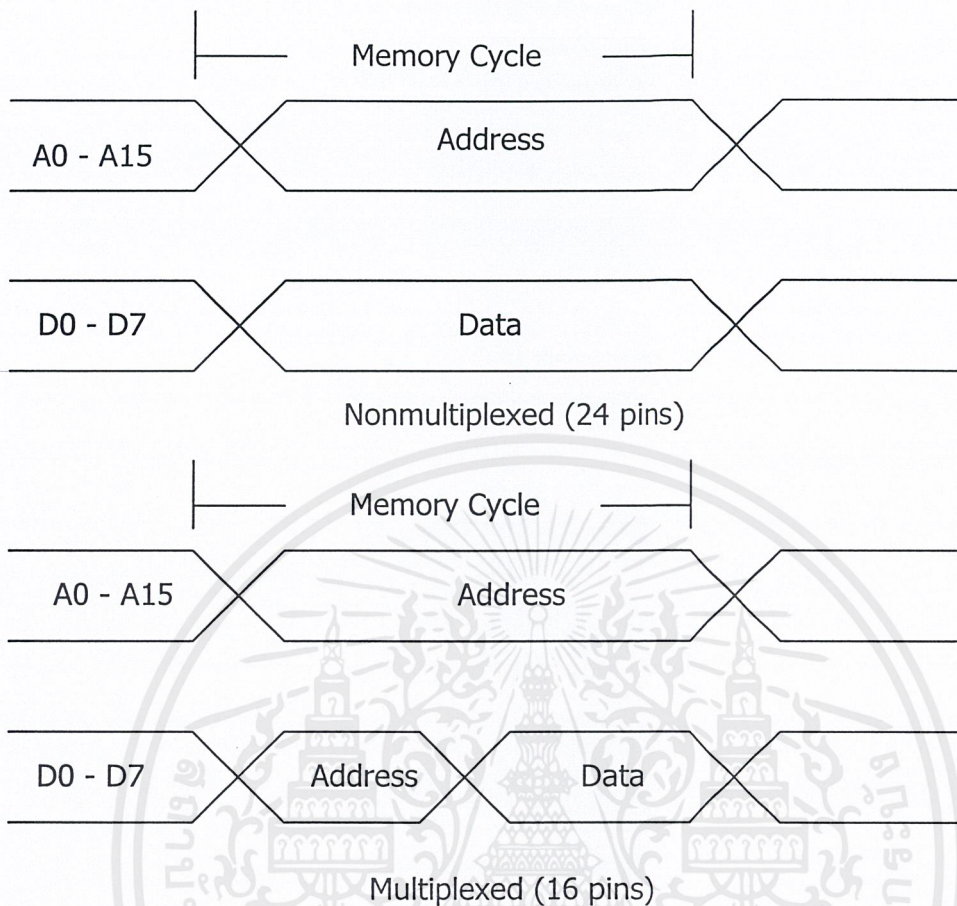
9) Power Control Register (PCON)

รีจิสเตอร์ PCON อยู่ที่ตำแหน่ง 87H ใช้หยุดการทำงานของ MCS-51 โดยจะหยุดจ่ายสัญญาณนาฬิกาให้ระบบ ทำให้ข้อมูลต่างๆ ภายใน MCS-51 ไม่มีการเปลี่ยนแปลง นอกจากนี้ยังลดพลังงานไฟฟ้าที่จ่ายให้ ใน MCS-51 ลงด้วย

2.2.7 หน่วยความจำภายนอก (External Memory)

MCS-51 ซึ่งสามารถอ้างหน่วยความจำข้อมูลภายนอกได้ 64K และยังสามารถถึงอ้างหน่วยความจำโปรแกรมภายนอกได้ 64K MCS-51 จะใช้พอร์ต 0 ในการอ้างตำแหน่งหน่วยความจำ 8 บิตล่าง และใช้พอร์ต 0 เป็นพอร์ตข้อมูล (DATA) ด้วยโดยใช้ขา ALE มาเป็น Latch ข้อมูลพอร์ต 0 และใช้พอร์ต 2 เป็นขาอ้างตำแหน่ง 8 บิต(รวมขาอ้างตำแหน่ง 16 เส้น ซึ่งอ้างได้ 64K)

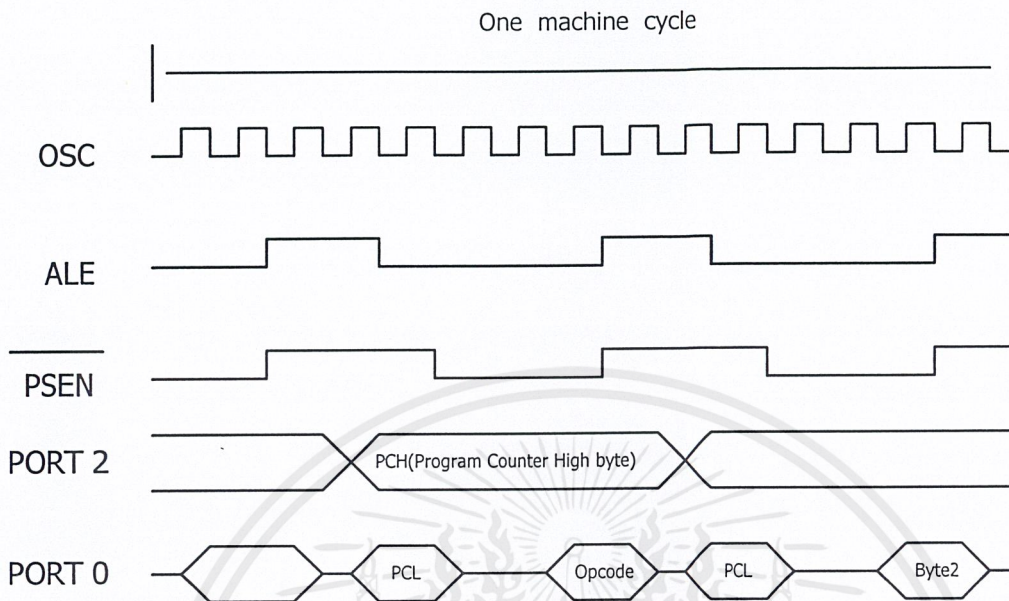
เนื่องจากพอร์ต 0 จะใช้งาน 2 หน้าทีในการติดต่อกับหน่วยความจำ จะใช้วิธี Multiplex ระหว่าง Address กับ Data พิจารณาจากรูป ถ้าต้องการติดต่อกับหน่วยความจำที่เก็บข้อมูล 8 บิต และเก็บได้ 64K จะต้องใช้สายสัญญาณ 24 เส้น คือ เป็นขา Address 16 เส้น และขาข้อมูล 8 เส้น ดังรูป แต่ถ้าใช้วิธี Multiplex คือ ใช้ขา A0-A7 เป็นขาข้อมูลด้วย คือ D0-D7 จะใช้สายสัญญาณเพียง 16 เส้นเท่านั้น จากรูป จะเห็นได้ว่าเมื่อต้องการติดต่อกับหน่วยความจำจะส่งสัญญาณ Address A0-A15 ออกมาก่อน 16 เส้น และเวลาต่อมาขา A0-A7 จะถูกเปลี่ยนเห็น D0-D7 ในการติดต่อกับหน่วยความจำภายนอกของ MCS-51 จะใช้วิธีนี้



รูปที่ 2.7 แผนผังกลุ่มสัญญาณที่ใช้อ่านข้อมูล

1) การติดต่อกับหน่วยความจำโปรแกรมภายนอก

ในการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอก MCS-51 จะส่งค่าตำแหน่งของหน่วยความจำออกไปก่อน ซึ่งค่าตำแหน่งจะเก็บอยู่ใน PC โดยส่งออกไปทางพอร์ต 0 และพอร์ต 2 จากนั้นเวลาต่อมา จะส่งขา ALE ให้เป็นลอจิก "0" เพื่อ Latch ขา Address ของ 8 บิตต่ำ คือ พอร์ต 0 จากนั้นจะส่งสัญญาณทาง PSEN ให้เป็นลอจิก "0" เพื่ออ่านข้อมูลซึ่งจะได้ Opcode เข้าไปทางขา Data Bus คือ พอร์ต 0 ไดอะแกรมเวลาอ่านข้อมูลจากหน่วยความจำภายนอกแสดงดังรูป 2.8



รูปที่ 2.8 แผนผังเวลาการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอก

2) การติดต่อกับหน่วยความจำข้อมูลภายนอก

หน่วยความจำข้อมูลภายนอก MCS-51 จะสามารถอ่านและเขียนได้ และในการติดต่อกับหน่วยความจำ ข้อมูลภายนอก MCS-51 จะส่งขา Address ออกไปทางพอร์ต 0 และ พอร์ต 2 จากนั้นจะส่งขา ALE เพื่อไป Latch Address 8 บิตค่า โดยการอ่านเขียนข้อมูลนั้นจะใช้ขา RD หรือ P3.7 และขา WR หรือ P3.6 ตามลำดับไคอะแกรมเวลาการอ่านและเขียนข้อมูลกับหน่วยความจำข้อมูลภายนอก

2.2.8 Reset Operation

การรีเซ็ตหรือเริ่มต้นทำงานใหม่ของ MCS-51 จะต้องให้ลอจิก “1” ที่ขา RST เป็นเวลา 2 Machine Cycles (1 Machine Cycles เท่ากับ 12 Clock) จากนั้นให้กลับเป็นลอจิก “0” การรีเซ็ตอาจทำได้โดยการใช้ สวิตช์กด หรือใช้วิธี Power-up โดยที่ใช้ตัว R-C มาต่อเป็นวงจรเมื่อ MCS-51 ถูกรีเซ็ต ค่ารีจิสเตอร์ต่างๆจะถูกกำหนดค่าตาราง โดย PC จะชี้ไปอยู่ตำแหน่งเริ่มต้น คือ 0000H เมื่อขา RST กลับเป็น “0” MCS-51 จะเริ่มทำโปรแกรมที่ตำแหน่งแรก

2.3 เทอร์โมคัปเปิล

2.3.1 ประวัติการค้นพบเทอร์โมคัปเปิล

ความเป็นมาในปี ค.ศ. 1821 นักวิทยาศาสตร์ชาวเยอรมัน โทมัส ซีเบ็ค (Thomas Seebeck) พบว่าเมื่อนำลวดโลหะ 2 เส้นที่ทำด้วยโลหะต่างชนิดกันมาเชื่อมต่อปลายทั้งสองข้างเข้าด้วยกัน ถ้าปลายจุดต่อทั้งสองมีอุณหภูมิต่างกัน ก็จะเกิดกระแสไฟฟ้าไหลวนในวงจรทั้งสอง ซึ่งปริมาณการไหลของกระแสไฟฟ้านี้ก็จะเปลี่ยนแปลงไปตามผลต่างของอุณหภูมิที่ปลายจุดต่อทั้งสอง และถ้าเปิดปลายจุดต่อด้านหนึ่งออก และจะทำให้เกิดแรงเคลื่อนไฟฟ้าที่ตำแหน่งปลายด้านเปิด ซึ่งแรงเคลื่อนไฟฟ้านี้เรียกว่า “ซีเบ็ค โวลเตจ”

2.3.2 กฎการเกิดแรงเคลื่อนไฟฟ้าจากความร้อน

- 1) กฎวงจรโลหะเดียว ในวงจรประกอบไปด้วยโลหะชนิดเดียวกัน จะไม่เกิดกระแสไฟฟ้าไหลวนในวงจรเมื่อเกิดความแตกต่างของอุณหภูมิ ณ ปลายจุดต่อทั้งสอง
- 2) กฎของโลหะแทรก ถ้ามีโลหะแทรกอยู่ในส่วนหนึ่งส่วนใดของวงจรหากจุดต่อของปลายโลหะแทรกทั้งสองมีอุณหภูมิเท่ากันแล้ว โลหะแทรกนี้จะไม่มีผลต่อแรงเคลื่อนไฟฟ้าที่เกิดขึ้น
- 3) กฎของอุณหภูมิแทรก แรงเคลื่อนไฟฟ้าที่เกิดขึ้นโดยเทอร์โมคัปเปิลชุดใดๆ ที่มีอุณหภูมิที่จุดต่อต่างกัน จะมีค่าเท่ากับผลบวกทางพีชคณิตของแรงเคลื่อนไฟฟ้าของเทอร์โมคัปเปิลชุดนั้น

2.3.3 ชนิดของเทอร์โมคัปเปิลแบบมาตรฐาน

1) เทอร์โมคัปเปิลแบบ J

เนื่องจากแพลตินัมเป็นธาตุที่มีราคาแพงเพื่อจะหาวัสดุที่มาแทนแพลตินัมจากการทดลองพบว่าโลหะผสมระหว่าง 60% ของทองแดง +40% ของนิกเกิล ที่เรียกว่า Constantan สามารถแก้ปัญหาได้ และต่อมากลายมาเป็นมาตรฐานแบบ J คุณสมบัติเหมาะสมสำหรับที่อุณหภูมิที่ไม่เกิน 760°C ไม่เหมาะสำหรับงานที่อุณหภูมิต่ำกว่า 0°C

2) เทอร์โมคัปเปิลแบบ K

สามารถทนอุณหภูมิได้สูงกว่าแบบ J และมีราคาถูกกว่า ทนอุณหภูมิได้ถึง 1260°C และที่อุณหภูมิต่ำถึง -250°C คุณสมบัติเด่นของเทอร์โมคัปเปิลแบบ K คือให้แรงดันเอาต์พุตสูง ข้อควรระวัง ไม่ควรใช้เทอร์โมคัปเปิลแบบ K สัมผัสโดยตรง

- 1) สภาพการทำงานที่เป็น Reducing หรือใช้สับกับระหว่าง Oxidizing และ Reducing
- 2) เทอร์โมคัปเปิล โดยเฉพาะที่สายลวดจะเกิดขึ้นอย่างรวดเร็ว
- 3) สภาพการทำงานที่เป็น Vacuum ยกเว้นในระยะช่วงสั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.4 เทอร์โมคัปเปิลและส่วนประกอบ

1) ตัวเทอร์โมคัปเปิล

1.1) ส่วนผสมของโลหะใช้ทำต้องไม่เปลี่ยนหรือสูญเสียธาตุส่วนผสมไปในเวลารวดเร็ว

1.2) จะต้องทำให้แรงเคลื่อนไฟฟ้าเอาต์พุตที่มีเสถียรภาพ(Stable)ในการใช้งานไม่เปลี่ยน

1.3) ต้องมีความแข็งแรงและทนทานทนต่อสภาพแวดล้อมในการใช้งานที่พิสูจน์ได้ โดยปกติจะถูกเชื่อมต่อกันทั้งสองเข้าด้วยกันเรียกว่าจุดต่อสำหรับวัด และร้อยด้วยฉนวนเพื่อป้องกันการลัดวงจรแบบที่มีใช้อยู่ทั่วไปในปัจจุบัน

2) กรอบโลหะป้องกันตัวเทอร์โมคัปเปิล

ลักษณะการประกอบตัวเทอร์โมคัปเปิลเข้ากับ Metal Sheath มี 3 วิธี แต่ละวิธีก็มีลักษณะการใช้งานเฉพาะตัวของมัน คือ แบบเปลือยจุดต่อ แบบที่ใช้ Metal Sheath เป็นจุดต่อเทอร์โมคัปเปิล และแบบไม่ใช้ Metal Sheath

3) ฉนวนของเทอร์โมคัปเปิล

ฉนวนของเทอร์โมคัปเปิลสำหรับกันตัวเทอร์โมคัปเปิลกับ Metal Sheath ส่วนใหญ่จะเป็นสารประเภทแมกนีเซียมออกไซด์ อะลูมิเนียมออกไซด์ หรือเบอริลเลียมออกไซด์ ลักษณะเดียวกับที่ใช้ในขดลวดทำความร้อน (Heater) ในปัจจุบัน คุณสมบัติของกระเบื้องฉนวนคือ เป็นตัวกันก๊าซ และ Vacuum ได้ดีเป็นพิเศษ เป็นฉนวนไฟฟ้าที่ดีแม้ที่อุณหภูมิสูงๆ ไม่ทำปฏิกิริยากับแพลตินัม ทังสเตน และโมลิบดีนัมที่เป็นส่วนผสมของตัวเทอร์โมคัปเปิล และสามารถทนอุณหภูมิใช้งานได้ถึง 1900°C

2.3.5 อาร์ทีดี

1) ประวัติความเป็นมาของอาร์ทีดี

ในปีเดียวกันกับซีเบ็คค้นพบกฎของพลังงานไฟฟ้าที่เกิดขึ้นจากความร้อน เซอร์ ฮัมฟรีย์ เดวี ก็ค้นพบว่า ค่าความต้านทานของลวดโลหะก็จะเปลี่ยนแปลงอุณหภูมิ อีก 50 ปีต่อมา เซอร์ วิลเลียม ซีเมนส์ ได้ทำการทดลองอย่างจริงจัง และพบว่า เส้นลวดแพลตินัมสามารถใช้เป็นตัวกลางวัดอุณหภูมิได้ และต่อมาได้กลายเป็นเครื่องมือวัดอุณหภูมิที่แพร่หลายในปัจจุบันตั้งแต่จุดแข็งตัวของออกซิเจน (-182.96°C) ไปจนถึงจุดแข็งตัวของแอนติโมนี (630.74°C)

2) โครงสร้างของอาร์ทีดี

ทำด้วยโลหะที่มีค่าหนึ่ง ซึ่งทำให้เกิดค่าความต้านทานที่ต้องการ ณ อุณหภูมิ 0°C ลวดโลหะนี้ จะพันอยู่บนแกนที่เป็นฉนวนไฟฟ้าและมีคุณสมบัติทนต่อความร้อน แกนที่ใช้เป็นสารประเภทเซรามิกหรือแก้ว สิ่งที่ต้องคำนึงถึงพิเศษในระหว่างกระบวนการผลิต คือ ขณะใช้งานขดลวดนี้ต้องทน

ต่ออุณหภูมิ และรวมถึงความสิ้นสะเทือนได้ ทั้งนี้เพราะเมื่อขดลวดได้รับความร้อนจะเกิดการขยายตัว
เมื่อเย็นจะหดตัว



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

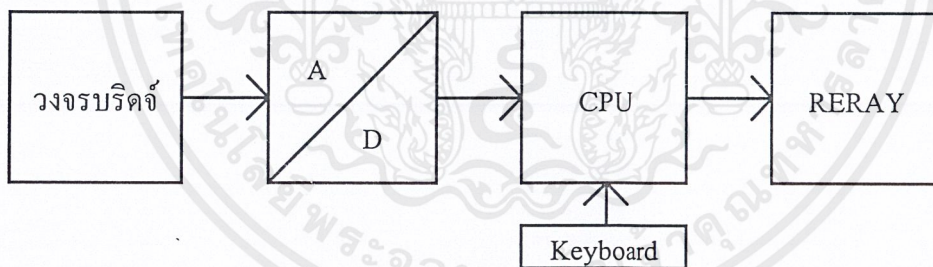
บทที่ 3

การออกแบบ การสร้าง และการทำงาน

ในส่วนของ การออกแบบการสร้างเครื่องมือวัดและเปรียบเทียบ Thermocouple (Measuring and Calibration Temperature) นี้แบ่งออกเป็น 2 ส่วน คือ ส่วนฮาร์ดแวร์และซอฟต์แวร์

3.1 ฮาร์ดแวร์ (Hardware)

ในส่วนของฮาร์ดแวร์มีการทำงานตามผังแสดงการทำงานดังรูปที่ 3.1 ซึ่งมีลักษณะการทำงานดังนี้ วงจรบริดจ์จะรับค่าการทำงานมาจากตัว RTD แล้งส่งไปยัง ADC 8 Bit เพื่อเปลี่ยนการทำงานจากสัญญาณแอนะล็อกที่รับมาจากตัว RTD เพื่อแปลงให้อยู่ในรูปสัญญาณดิจิทัล จะส่งผลไปยัง CPU โดยใช้ MCS-51 เป็นตัวประมวลผลการทำงาน และใช้ Key board ทำการส่งคำสั่งจากภายนอก เพื่อสั่ง CPU ให้ทำงานตามที่คำสั่งภายนอกสั่งมา เมื่อ CPU ประมวลผลตามที่ต้องการแล้ว ก็จะส่งผลออกไปยัง Relay จะทำการตัดต่อ Heater และพัดลมระบายอากาศ ถ้าจะทำให้อุณหภูมิลดลงเร็ว เราก็สั่งให้พัดลมทำงานตามโปรแกรมคำสั่ง

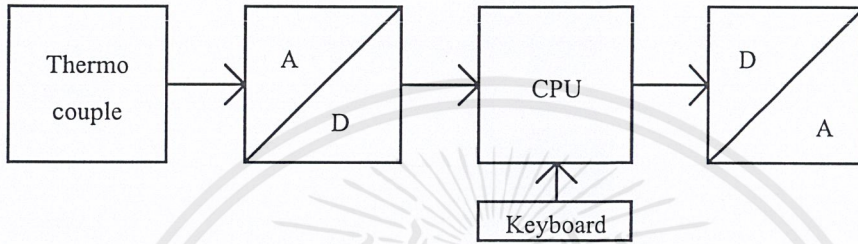


รูปที่ 3.1 ผังการทำงานของเครื่องมือวัดและเปรียบเทียบ Thermocouple

ในส่วนของ การสร้างอุณหภูมิ

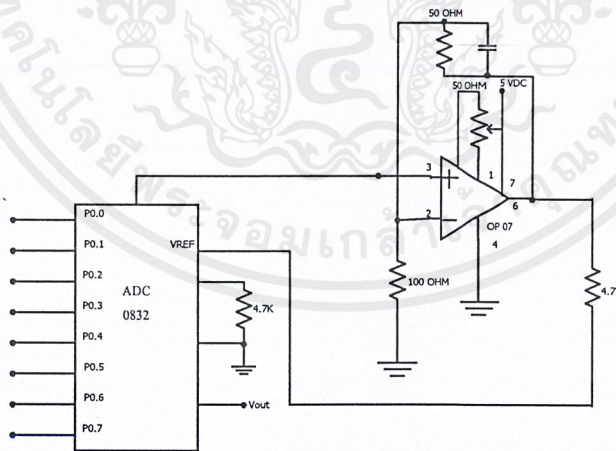
ในส่วนของฮาร์ดแวร์ตัวที่ 2 จะทำงานตามผังแสดงการทำงานดังรูปที่ 3.2 มีลักษณะการทำงานดังนี้ เรารับค่าอุณหภูมิมาจากตัว Thermocouple แล้งส่งไปยัง ADC 8 Bit เพื่อเปลี่ยนสัญญาณแอนะล็อกมาเป็นสัญญาณดิจิทัล เพราะ Thermocouple จะส่งสัญญาณมาในรูปของสัญญาณแอนะล็อกเราจึงทำการแปลงสัญญาณก่อน เพื่อจะได้ส่งค่าไปยังตัว CPU ประมวลผล และเมื่อส่งมายัง CPU ประมวลผล

เรียบร้อยแล้ว CPU ก็จะทำการประมวลผลตามที่เราโปรแกรมไว้แล้ว ถ้าเกิดการผิดพลาดเราก็จะใช้ Key board ในการแก้คำสั่งที่กระทำผิดไป เมื่อแก้ไขหรือทำคำสั่งใหม่เรียบร้อยแล้ว CPU ประมวลผลจะส่งค่าไปยัง DAC 8 Bit ตัว DAC 8 Bit ก็จะทำการแปลงสัญญาณดิจิตอลมาเป็นสัญญาณแอนะล็อก โดยส่งสัญญาณมาเป็นแอนะล็อก โดยสัญญาณแอนะล็อกที่ส่งมาจะมีหน่วยเป็นมิลลิโวลต์ (mV) ตามที่เราโปรแกรมไว้ใน CPU โดยเทียบกับตารางค่าความเป็นจริงของอุณหภูมิ



รูปที่ 3.2 ผังการทำงานของเครื่องมือวัดและเปรียบเทียบ Thermocouple ในส่วนของตัว Calibration

3.1.1 การออกแบบวงจร A/D 8 Bit

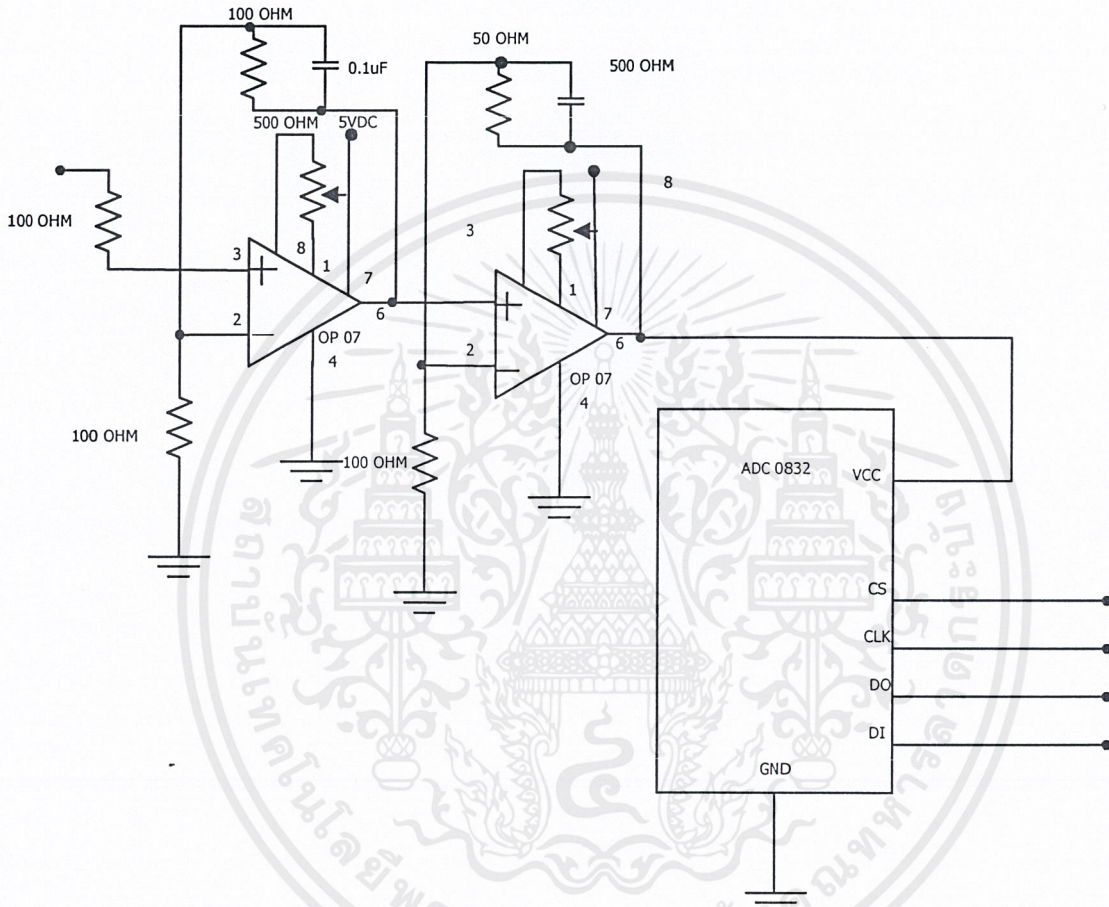


รูปที่ 3.3 ADC 8 Bit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 การออกแบบวงจร D/A 8 Bit

ในโครงการนี้ได้ใช้การประมวลผลทางดิจิทัลรวมกับการประมวลผลโดยไมโคร จึงเลือกใช้การ์ดแปลงสัญญาณดิจิทัลเป็นแอนะล็อก

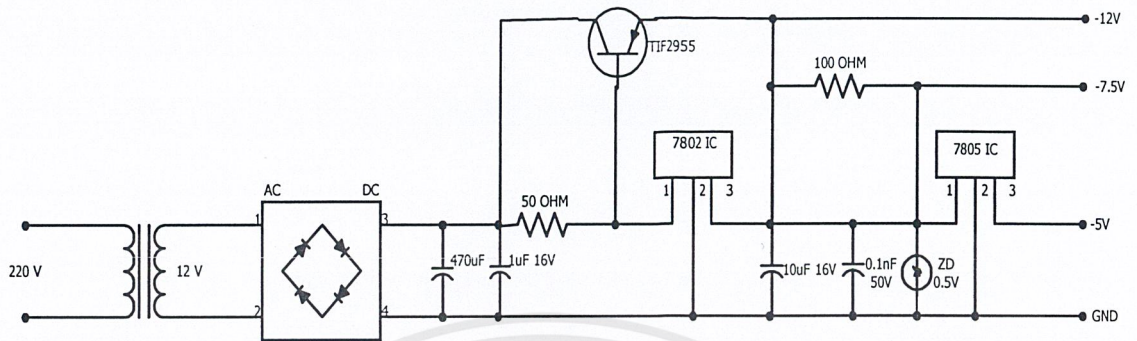


รูปที่ 3.4 การใช้งาน D/A 8 Bit

3.1.3 วงจรแหล่งจ่ายตัวสร้างอุณหภูมิ

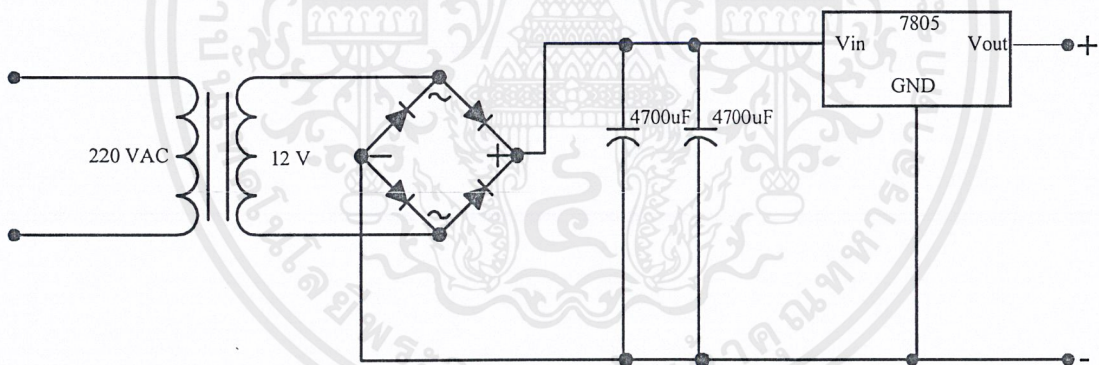
วงจรแหล่งจ่ายไฟฟ้ากระแสตรงในโครงการนี้ใช้ IC ทรานซิสเตอร์ 78XX ซึ่งทำหน้าที่รักษาระดับแรงดัน ในโครงการนี้ใช้แหล่งจ่ายไฟฟ้ากระแสตรง 3 ชุด ประกอบด้วยแรงดัน +5V +7.5V และ +12V ใช้ IC เบอร์ 7812 และ 7805 ซึ่งมีลักษณะวงจรดังรูปที่ 3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 วงจรแหล่งจ่ายตัวสร้างอุณหภูมิ

3.1.4 วงจรแหล่งจ่ายตัว Calibration

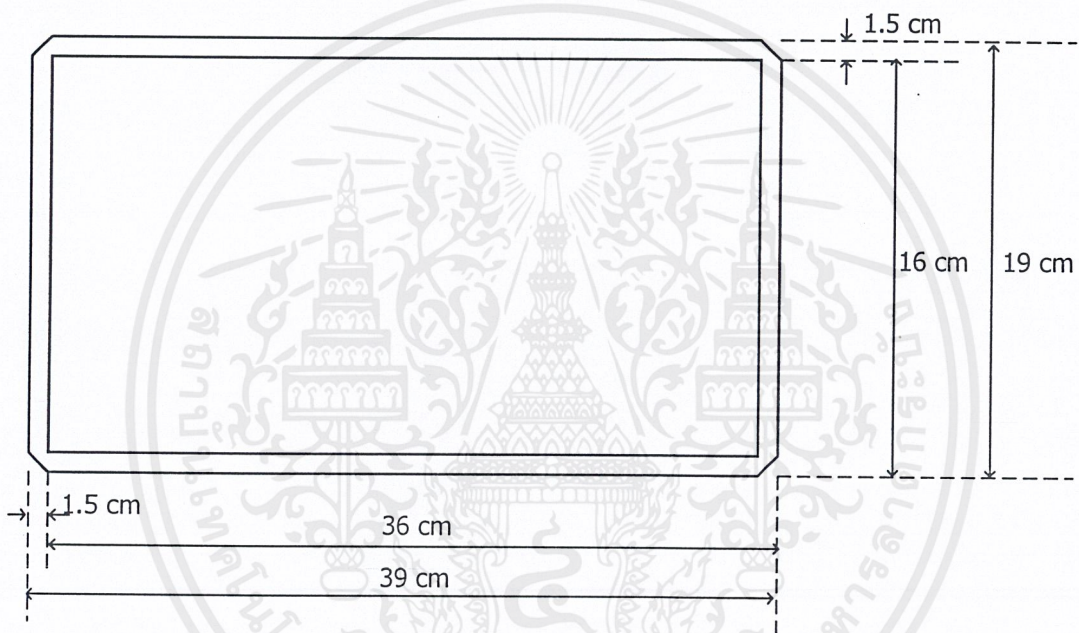


รูปที่ 3.6 วงจรแหล่งจ่ายตัว Calibration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

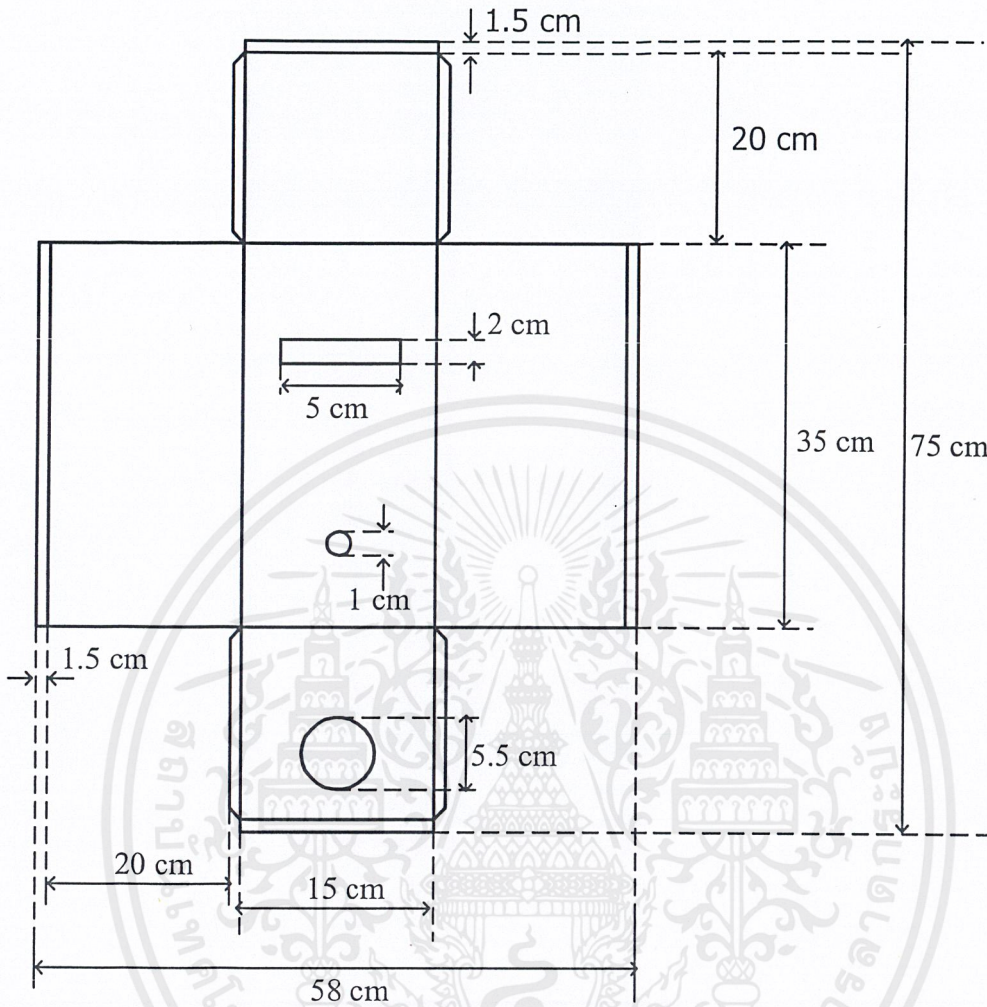
3.1.5 การออกแบบกล่องตัวสร้างอุณหภูมิตัว

ในการออกแบบทำกล่องใส่ตัวสร้างอุณหภูมิตัวหรือห้องสร้างอุณหภูมิตัว ในส่วนที่ 1 นี้ จะเป็นด้านล่างของกล่อง มีรูปร่างลักษณะดังรูปที่ 3.7 จะมีขนาดตามที่เรากำหนดขึ้นมาเพื่อความเหมาะสมของชิ้นงาน จะมีความกว้างของกล่องเท่ากับ 19 เซนติเมตร และความยาวของกล่องเท่ากับ 39 เซนติเมตร จากนั้นทำการเขียนแบบในกระดาษเขียนแบบ เสร็จแล้วก็นำเหล็กแผ่นขนาดบางมาวาง แล้วจากนั้นใช้ปากกาวาดแบบลงบนแผ่นเหล็ก ตามแบบรูปที่ 3.7 เมื่อวาดแบบเสร็จแล้ว ทำการตัดส่วนที่ไม่ใช่ ออก แล้วทำการพับ



รูปที่ 3.7 รูปด้านล่างของกล่องสร้างอุณหภูมิตัว

ในส่วนที่ 2 เป็นส่วนของตัวกล่องอุณหภูมิตัวหรือห้องสร้างอุณหภูมิตัว มีรูปร่างลักษณะดังรูปที่ 3.8 โดยส่วนตัวของกล่องนี้จะมีขนาดความกว้างเท่ากับ 58 เซนติเมตร และความยาวเท่ากับ 78 เซนติเมตร จากนั้นก็วาดแบบลงบนกระดาษก่อน เมื่อเสร็จแล้วก็นำเหล็กแผ่นมาเขียนแบบตามรูปที่ 3.8 จากนั้นตัดส่วนที่ไม่ใช่ทิ้งไป ทำการพับตามรอยเส้นด้านในให้เป็นรูปร่างขึ้นมา แล้วเจาะรูตามที่เขียนแบบไว้โดยละเอียด เมื่อพับเสร็จแล้วก็ใช้สว่านเจาะรู เพื่อจะใช้รื้อเวทีย้ำให้ติดกันเป็นรูปกล่องตามที่เรากำหนดไว้ เมื่อเสร็จแล้ว ก็ตกแต่งให้สวยงาม



รูปที่ 3.8 ส่วนตัวกล่องสร้างอุณหภูมิ

3.2 ซอฟต์แวร์

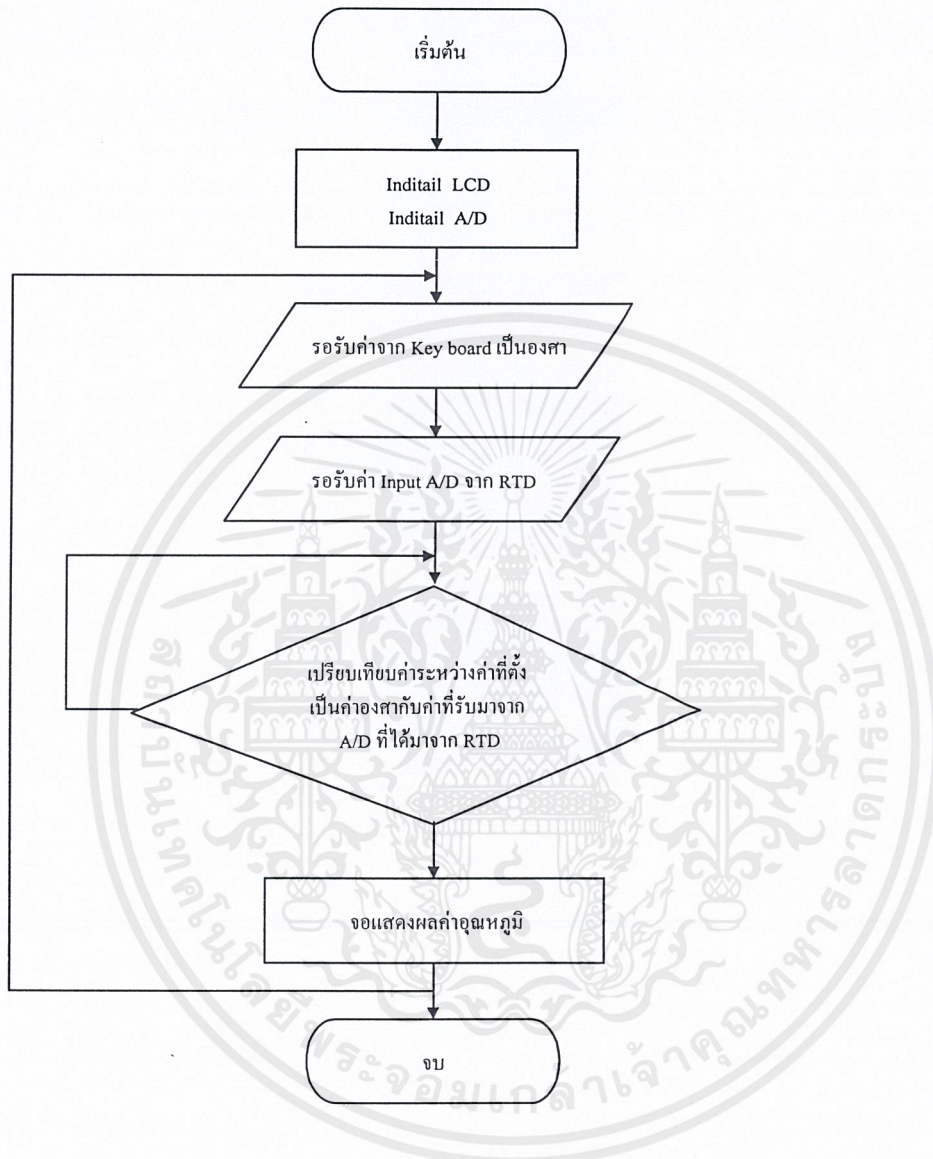
ในส่วนซอฟต์แวร์ จะแบ่งออกเป็น 2 ส่วนใหญ่ๆ คือ ส่วนของตัวสร้างอุณหภูมิและส่วนของตัว Calibration

3.2.1 ส่วนของตัวสร้างอุณหภูมิ ในส่วนนี้จะประกอบด้วย 2 ส่วน คือ

- 1) ส่วนการควบคุม Heater เพื่อสร้างอุณหภูมิ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

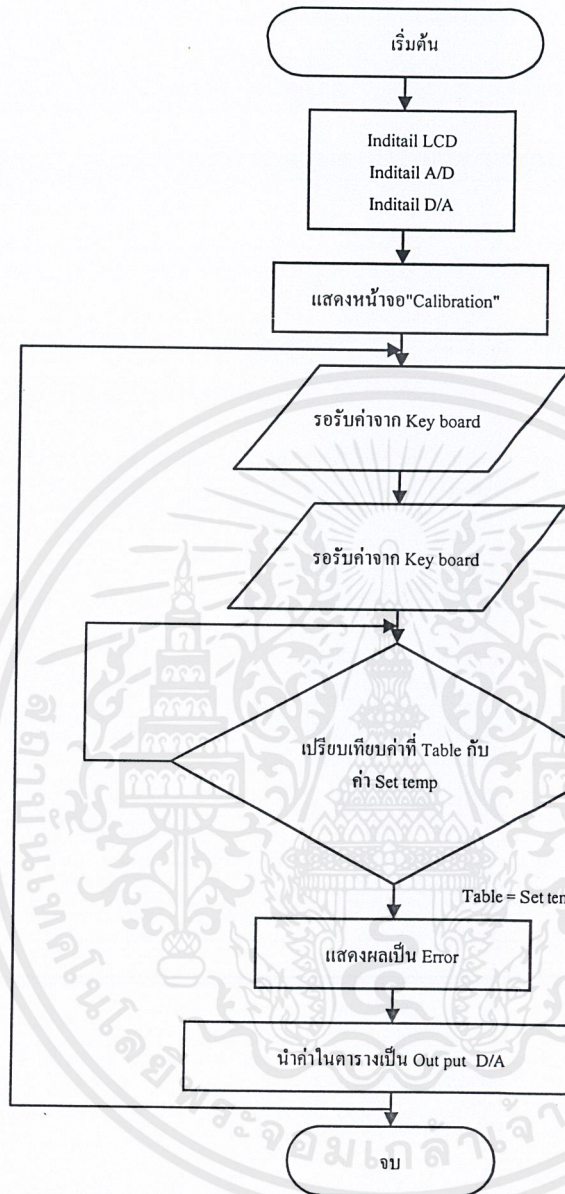
2) ส่วนแสดงผล ในส่วนนี้จะแสดงผลค่าของอุณหภูมิที่เรากำหนดไว้



รูปที่ 3.9 แผนผังการทำงานของตัวสร้างอุณหภูมิ

3.2.2 ส่วนของตัว Calibration ในส่วนนี้จะทำการ Calibration ค่าอุณหภูมิที่เรากำหนดโดยใช้ตารางปรับเทียบค่าอุณหภูมิ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 ผังแสดงการทำงานของส่วนของตัว Calibration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 การทดลองและการทดสอบ

4.1.1 อุปกรณ์ที่ใช้ในการทดลอง

- 1) วงจรบริดจ์
- 2) ดิจิตอลมิเตอร์
- 3) RTD
- 4) Thermocouple
- 5) A/D 8 Bit

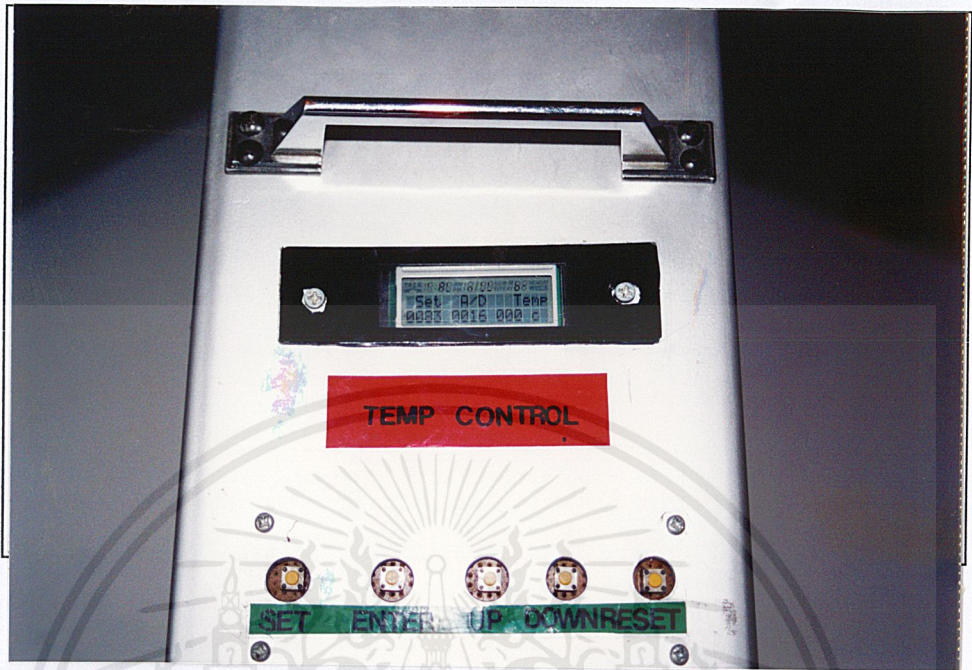
4.1.2 ลำดับขั้นการทดลอง

- 1) นำวงจร Bridge มาต่อเข้ากับตัว RTD ซึ่งเป็นตัวตรวจสอบอุณหภูมิของ Heater
- 2) ทำการสร้างอุณหภูมิที่ต้องการตามลำดับ
- 3) ทำการปรับค่าความต้านทานที่ทำให้ค่าสัญญาณออกมาตามที่ต้องการ
- 4) นำตัว Thermocouple มาเทียบที่ตัวสร้างอุณหภูมิ
- 5) ใช้มัลติโวลต์มิเตอร์มาวัด
- 6) นำผลที่ได้จากการวัดมาเทียบตาราง Thermocouple
- 7) ทำการแก้ไขตัวสร้างอุณหภูมิให้ตรงตามตาราง
- 8) บันทึกผลการทดลอง

ตารางที่ 4.1 ผลการทดลอง A/D 8 Bit

อุณหภูมิ (องศา C)	ตำแหน่ง Step	แรงดันไฟฟ้า(โวลต์)
400	185	3.6260
380	186	3.6456
360	187	3.6652
340	188	3.6848
320	189	3.7044
300	190	3.7240
280	191	3.7436
260	192	3.7632
240	193	3.7828
220	194	3.8024
200	195	3.8220
180	196	3.8416
160	197	3.8612
140	198	3.8808
120	199	3.9004
100	200	3.9200
80	201	3.9396
60	202	3.9592
40	203	3.9788
*	204	3.9984
20	205	4.0180

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

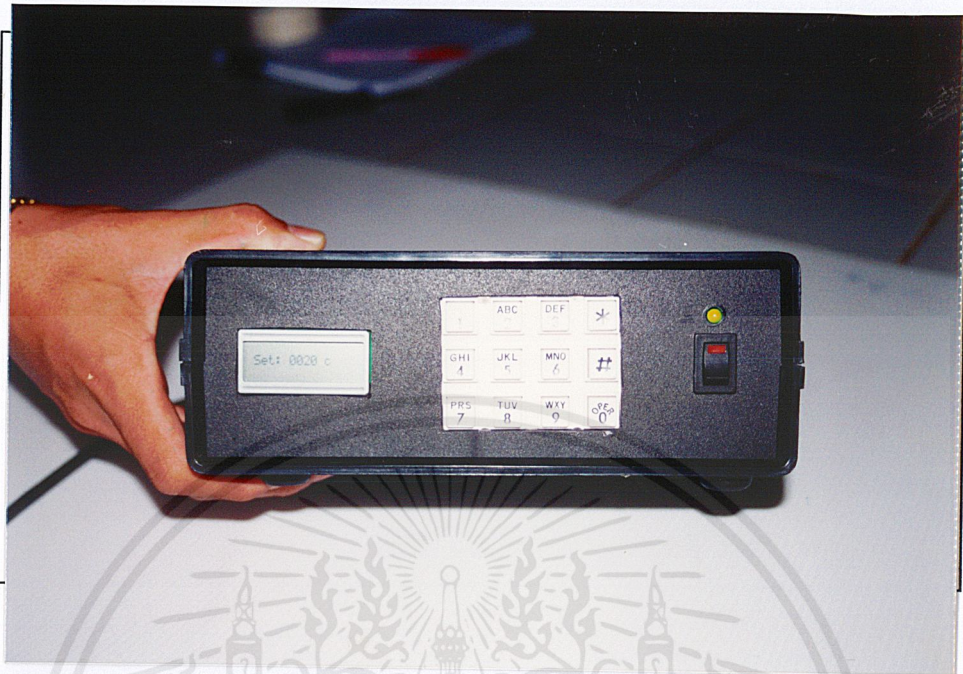


รูปที่ 4.1 รูปการทำงานของ Temp control



รูปที่ 4.2 รูปการทำงานของเครื่อง Calibration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 ลักษณะของการ Set ของเครื่อง Calibration



รูปที่ 4.4 ลักษณะของ Display ของเครื่อง Calibration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทสรุป ปัญหา แนวทางแก้ไข และการพัฒนา

5.1 บทสรุป

การจัดทำเครื่องตรวจสอบ Thermocouple จัดทำขึ้นเพื่อศึกษาการทำงานของการทำงานของเครื่องตรวจสอบตัว Thermocouple ว่ายังมีการทำงานคืออยู่หรือไม่ ซึ่งในการตรวจสอบนี้ก็จะทำโดยง่าย โดย การตั้งค่าอุณหภูมิตามที่ต้องการตรวจสอบ แล้วนำค่าที่ได้มาบันทึกในกราฟแล้วเทียบการเปลี่ยนแปลงของตัว Thermocouple ว่าคุณลักษณะเป็นอย่างไร มีการเกิดฮิสเตอร์รีซิสหรือไม่ ซึ่งการเกิดฮิสเตอร์รีซิสนี้จะไม่เป็นผลที่ติดต่อเครื่องมือวัดอุณหภูมิ ในโครงการนี้จะมีตัวที่สร้างอุณหภูมิด้วย และตัวสร้างอุณหภูมิยังใช้หลักการเซนเซอร์ของตัว RTD เป็นตัวเซนเซอร์โดยหลักการของตัว RTD ก็คือเมื่อมีอุณหภูมิสูงค่าความต้านทานก็จะสูงตามด้วยการทดสอบตัว Thermocouple นี้จะมีการลำดับขั้นตอนที่กว้างมากแต่ก็สามารถที่จะทำการตรวจสอบได้โดยไม่มีปัญหา แต่ข้อบกพร่องนี้ก็มาจากตัวสร้างอุณหภูมิซึ่งมีความละเอียดน้อย ในการตรวจสอบนี้ไม่สามารถที่จะไปแก้ไขตัว Thermocouple ได้ในโครงการนี้สามารถตรวจสอบได้เพียงอย่างเดียวในการติดตั้งตัวเซนเซอร์ในตัวสร้างอุณหภูมิ ก็จะติดตั้งด้านล่างของตัว Heater ซึ่งจะเป็นลักษณะแท่งอลูมิเนียมกลมจะติดตั้ง RTD ไว้ด้านล่างและด้านบนก็จะเจาะรูเพื่อเป็นที่ใส่ตัว Thermocouple ที่ต้องการตรวจสอบว่าสามารถที่จะใช้งานต่อไปได้อีกหรือไม่ ซึ่งการจัดทำตัวตรวจสอบ Thermocouple นี้ก็เพื่อเราจะได้ไม่ต้องใช้อุปกรณ์ที่นำเข้ามาจากต่างประเทศซึ่งเป็นการสิ้นเปลืองทางด้านขนส่งมากและภายในตัวนี้ยังมีตารางของตัว Thermocouple อยู่อีกด้วย แต่ข้อบกพร่องของอุปกรณ์ตัวนี้คือสามารถตรวจสอบได้แค่สองแบบคือ J,K เท่านั้น แต่แบบมาตรฐานนี้จะมีถึง 7 แบบ

เครื่องตรวจสอบ Thermocouple นี้สามารถทำการตรวจสอบได้คือ

- 1) สามารถบอกค่าความผิดพลาดของตัว Thermocouple ได้
- 2) สามารถบอกค่าที่มาตรฐานของ Thermocouple ได้
- 3) สามารถสร้างอุณหภูมิตามที่ต้องการได้
- 4) สามารถแสดงผลบนจอ LCD ได้
- 5) สามารถนำไปประยุกต์ใช้เป็นการในวิชาเครื่องมือวัดได้

5.2 ปัญหาและแนวทางแก้ไข

การจัดทำเครื่องมือตรวจสอบ Thermocouple นี้ในปริิณญาณิพนธ์นี้ได้พบปัญหาที่เกิดขึ้นดังนี้

1) ปัญหาด้านฮาร์ดแวร์ ในตัวสร้างอุณหภูมิมีปัญหาที่คือตัว Heater ที่นำมาสร้างอุณหภูมิที่ใช้มีการอุณหภูมิที่เร็วมากแต่ในการลดอุณหภูมิจะมีการลดที่ช้ากว่าตอนแรกเราทำการติดพัดลมโดยเป่าขึ้นแต่ตำแหน่งการติดตั้งพัดลมไม่สามารถติดได้


แนวทางแก้ไข การคิดในลักษณะควบคุมออกและการลดของอุณหภูมิก็สามารถลดได้เร็วกว่าการเป่าลมด้วยอีกปัญหาหนึ่ง ก็คืออุณหภูมิที่สร้างจะมีผลทำให้ส่วนควบคุมในวงจรอิเล็กทรอนิกส์เกิดการเสียหาย การแก้ปัญหาโดยการกันเป็นห้องแล้วนำเอาฉนวนกรวด้านในด้วย

2) ปัญหาด้านซอฟต์แวร์ ต้องทำการศึกษาการทำงานของ CS5526 เป็น IC ที่ทำการแปลงสัญญาณแอนะล็อกเป็นดิจิทัลการแก้ไขก็คือการไปหาข้อมูลจากอินเทอร์เน็ต และไปขอโปรแกรมจาก SILA แต่โปรแกรมที่ให้มาเป็นภาษาซี การเขียนโปรแกรมในตอนนี้เขียนเป็นภาษาแอสซิมบลีก็เลยต้องทำการศึกษาจากภาษาที่ให้มาที่เป็นภาษาซีให้เป็นภาษาแอสซิมบลี เพราะไม่เคยเขียนภาษาซีเลยอีกตัวหนึ่งคือตัว DAC 20 Bit ก็เช่นกัน ซึ่ง IC ที่นำมาใช้นี้เป็น IC ที่ไม่แพร่หลายนักจึงไม่ค่อยมีใครนำมาใช้ด้วยการสั่งงานด้วยโปรแกรมก็มีความยุ่งยากมากพอสมควร การสั่งซื้อก็เช่นกันในประเทศไม่ค่อยมีใครนำมาจำหน่ายจึงต้องมีการสั่งซื้อจากต่างประเทศ บางอย่างเช่น DAC 20 Bit

แนวทางการแก้ไข ทำการหาข้อมูลที่เกี่ยวข้องกับตัวแปลงสัญญาณแอนะล็อกเป็นดิจิทัลและดิจิทัลเป็นแอนะล็อกโดยการหาทางอินเทอร์เน็ต สอบถามอาจารย์ที่ปรึกษาและไปขอโปรแกรมจากผู้ขายและผู้เคยการใช้งานตัวอุปกรณ์เหล่านี้

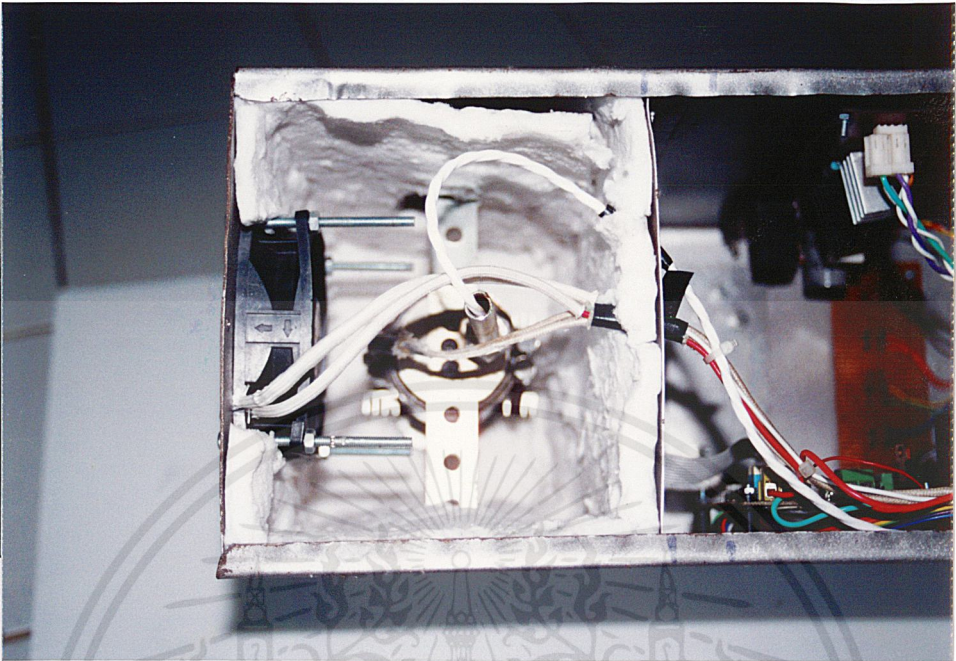
5.3 แนวทางการพัฒนา

อุปกรณ์การวัดและปรับเทียบ ตัวนี้ถ้าต้องการให้มีความสามารถในการทำงานมากขึ้นจาก 2 type มาเป็น 7 type ตามมาตรฐานของอุปกรณ์การวัดประเภทนี้ ส่วนทางด้านค่าความคลาดเคลื่อนต่างๆ ก็ควรจะลดน้อยลงเพราะงานในการปรับเทียบนั้นมีความจำเป็นและสำคัญมาก เมื่อมีความคลาดเคลื่อนที่ต่ำแล้วนั้นก็จะเป็นที่ยอมรับในงานอุตสาหกรรม

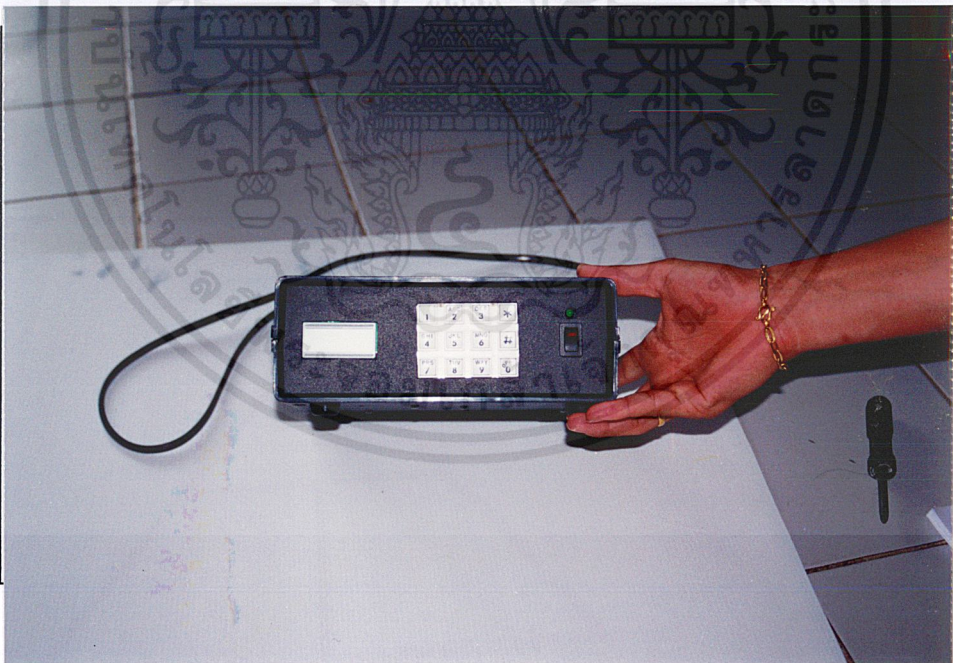


ภาคผนวก ก
รูปต้นแบบของเครื่องวัดและปรับเทียบ Thermocouple

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

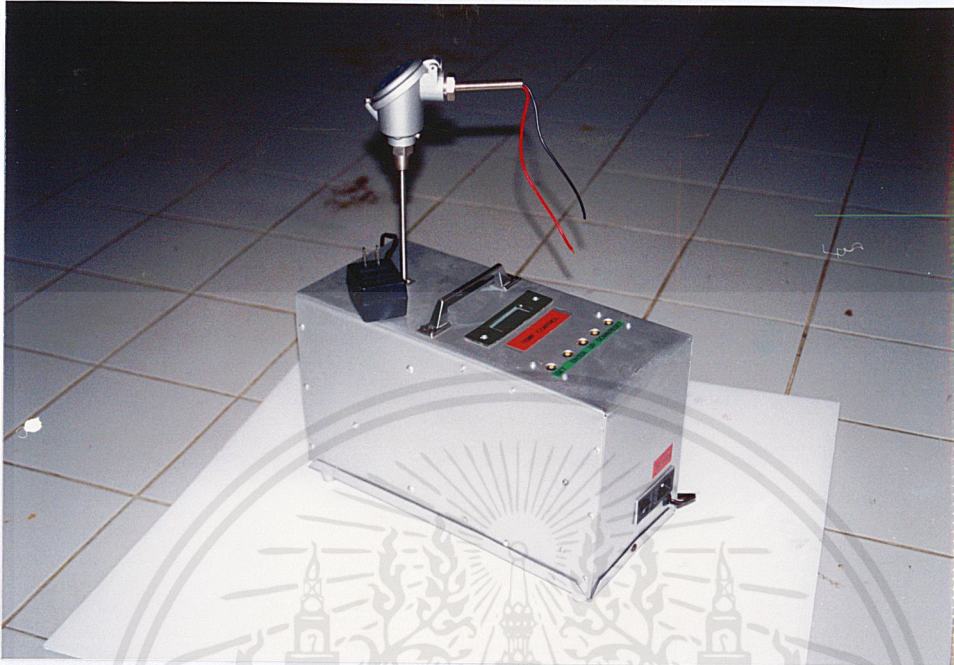


รูปที่ ก.1 ส่วนของห้อง Heater



รูปที่ ก.2 เครื่อง T.C Calibration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.3 เครื่อง Temp Control



รูปที่ ก.4 การต่อใช้งานเพื่อทำการ Calibration Thermocouple

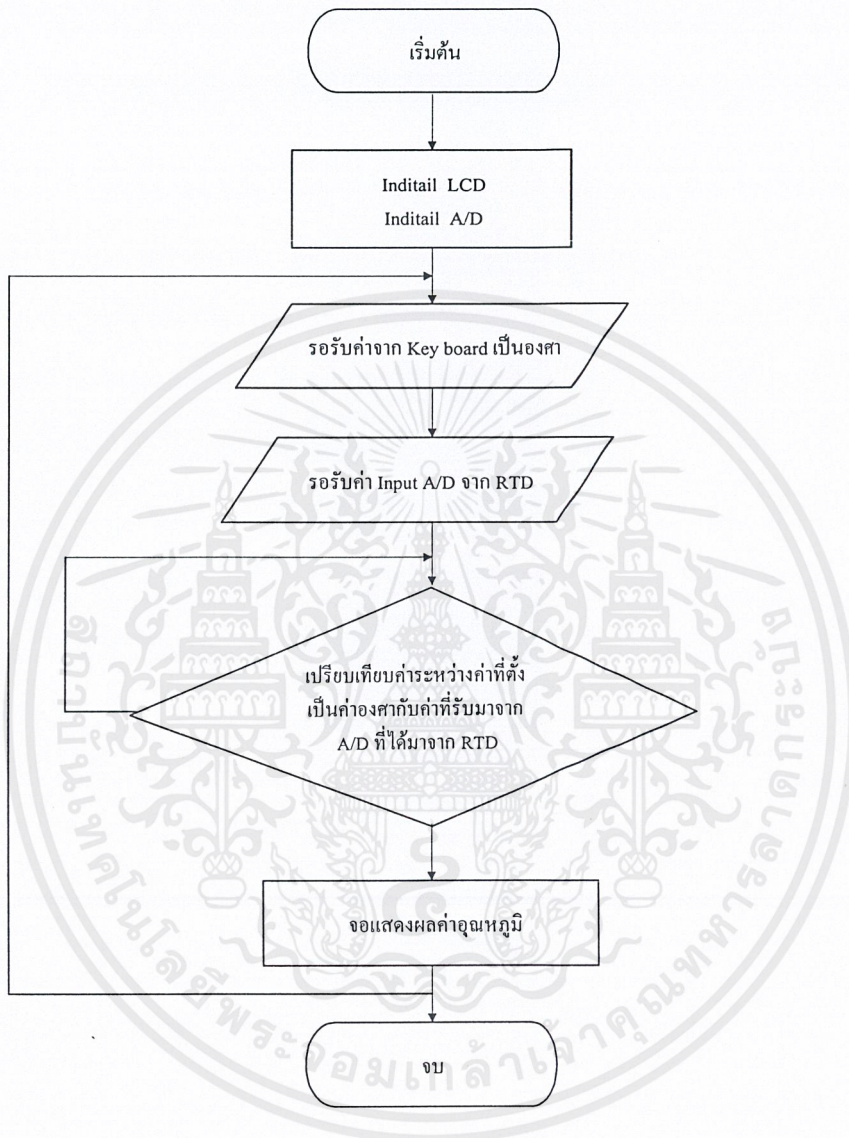
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ข
วงจรถวายงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

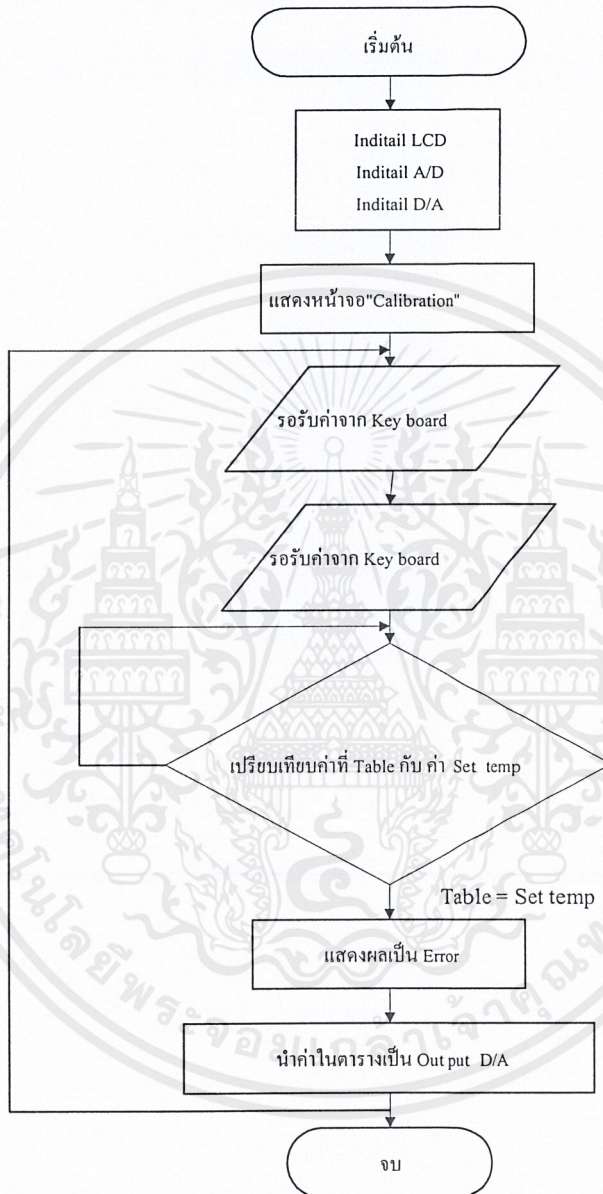
แผนผังการทำงานของเครื่องวัดและเปรียบเทียบ Thermocouple



รูปที่ ข.1 แผนผังการทำงานของตัวสร้างอุณหภูมิ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนผังการทำงานของตัว Calibration



รูปที่ ข.2 แผนผังการทำงานของตัว Calibration

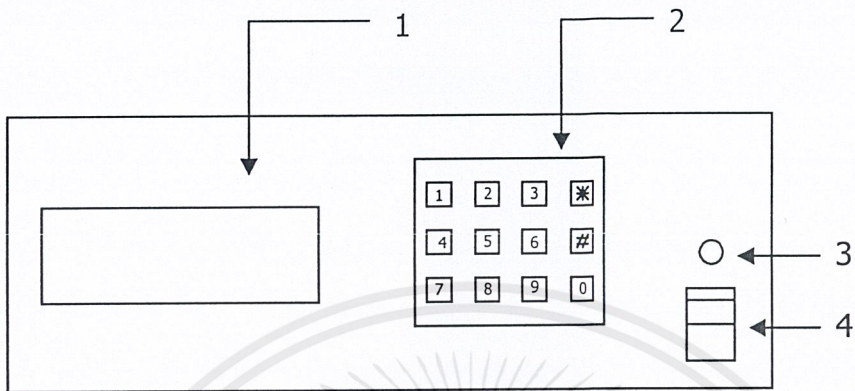
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



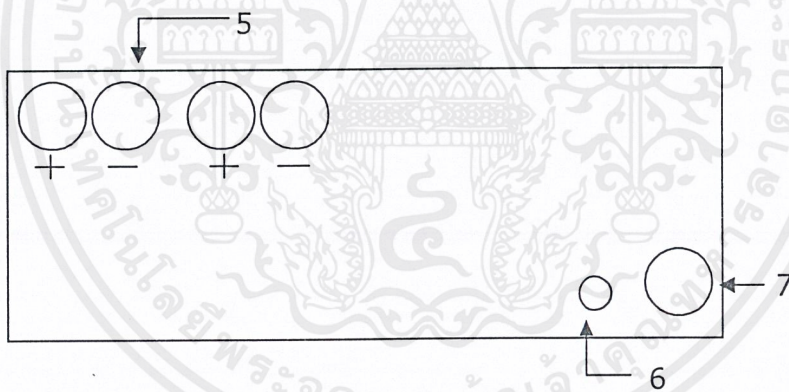
ภาคผนวก ค
คู่มือการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ส่วนประกอบของเครื่อง T.C. Calibration



รูปที่ ค.1 ส่วนประกอบด้านหน้า



รูปที่ ค.2 ส่วนประกอบด้านหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเลข	ความหมาย
1	จอแสดงผล (LCD)
2	คีย์บอร์ด
3	ไฟแสดงการทำงาน (LED)
4	สวิตช์เปิด-ปิด
5	จุดต่อ TC Input และ Output
6	เต้าเสียบ 220 V 50 Hz
7	ฟิวส์

คู่มือการใช้งานของตัวอุปกรณ์ Thermocouple Calibration Temperature

การใช้งานจะแบ่งเป็น 2 ส่วนคือ

1. ส่วนตัวสร้างอุณหภูมิ
2. ส่วนของตัว Calibration

1. ส่วนของตัวสร้างอุณหภูมิ

การใช้ตัวสร้างอุณหภูมิสามารถนำตัวสร้างอุณหภูมิไปใช้งานกับงานอื่นๆ ได้นอกเหนือจากการใช้งานกับตัว Calibration การใช้งานของตัวสร้างอุณหภูมิตั้งกับตัว Calibration คือ

- 1) เสียบปลั๊กในส่วนของวงจรควบคุม
- 2) เสียบปลั๊กในส่วนของวงจร Power
- 3) หน้าจอแรกที่ปรากฏบนจอ LCD
- 4) ทำการตั้งค่าอุณหภูมิที่ต้องการโดยการกดสวิตช์ Set
- 5) ทำการตั้งค่าโดยใช้สวิตช์ Up และ Down ทำการตั้งค่าอุณหภูมิโดยค่าของอุณหภูมิ

และจะเป็นหน่วย องศาเซลเซียส

- 6) เมื่อตั้งค่าองศาได้ตามที่ต้องการแล้วให้กดสวิตช์ Enter เพื่อตกลงการสร้างอุณหภูมิ
- 7) หน้าจอก็จะแสดงผลค่าของอุณหภูมิที่สร้างขึ้นจนถึงค่าอุณหภูมิที่ตั้งไว้
- 8) เมื่อต้องการตั้งค่าอุณหภูมิใหม่ก็ให้กดสวิตช์ Set แล้วทำซ้ำตั้งแต่ขั้นตอนที่ 1.5 ลงมา
- 9) เมื่อต้องการจะหยุดการทำงานให้กดสวิตช์ Reset แล้วชักปลั๊กออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ส่วนของตัว Calibration

การใช้งานของตัว Calibration เป็นตัวตรวจสอบตัว Thermocouple ว่ายังสามารถใช้งานต่อไปได้อีกหรือไม่ การใช้งานคือ

- 1) เสียบปลั๊กแล้วเปิด Switch Power
- 2) หน้าจอแรกจะแสดงชื่อของตัว Calibration “ T.C. THERMOCOUPLE ”
- 3) ทำการกดปุ่มใดๆ เพื่อให้อยู่ใน Mode ของการ Calibrated
- 4) ทำการตั้งค่าอุณหภูมิให้สัมพันธ์กับตัวสร้างอุณหภูมิ
- 5) คูที่หน้าจอบรรทัดที่สองจะแสดงค่าความคลาดเคลื่อน
- 6) แล้วทำการเพิ่มค่าอุณหภูมิสูงขึ้นเป็น Step แล้วดูค่าความคลาดเคลื่อน
- 7) แล้วบันทึกผลว่ามีความคลาดเคลื่อนมากน้อยเพียงใดตัว Thermocouple ยังสามารถใช้งานได้ต่อไปหรือไม่โดยการดูค่าความคลาดเคลื่อน
- 8) เมื่อทำการใช้งานเสร็จเรียบร้อยแล้วก็ปิด Switch Power



ภาคผนวก ง

โปรแกรมการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการทำงาน Temp Control

```

;*****
;
;   PROJECK
;*****COMMAND*****
CS      BIT   P1.0
SDI     BIT   P1.1
SDO     BIT   P1.2
SLK     BIT   P1.3
;*****WTK BOARD*****
DAT_595 BIT   P0.0
OUT_ENA_595 BIT P0.1
LACH_595 BIT   P0.2
CLK_595  BIT   P0.6

LED2     BIT   P0.4
AC_LED   BIT   P1.4
WD       BIT   P1.5
SPK      BIT   P2.0
;*****RTC*****
RSTBIT   BIT   P2.5
DATABIT  BIT   P2.6
XCLKBIT  BIT   P2.7
;*****LCD*****
CS_LCD   BIT   P3.3
RS_LCD   BIT   P3.4

        ORG 20H
FLAG:    DS    1
BUF_959  DS    2
CNT      DS    1

```

```
,*****24C16*****
```

```
COMMAND:
```

```
LF      EQU  0AH
```

```
DR      EQU  0DH
```

```
LINE1   EQU  00H
```

```
LINE2   EQU  0C0H
```

```
,*****INT*****
```

```
ORG 0000H
```

```
SJMP P_RESET
```

```
ORG 0BH
```

```
LJMP TIMER0_INT
```

```
ORG 23H
```

```
LJMP SERIAL_INT
```

```
P_RESET:
```

```
MOV PCON,#08H
```

```
MOV TMOD,#22H
```

```
MOV TH1,#0FAH
```

```
MOV TL1,#0FAH
```

```
SETB TCON,6
```

```
MOV SCON,#052H
```

```
MOV SP,#(STACK-1)
```

```
MOV TH0,#256-192
```

```
MOV TL0,#256-192
```

```
INTRO:  CALL PRINTSTR
```

```
DB 'SET A/D C',00
```

```
CALL INT_LCD
```

```
SETB TR0
```

```
SETB ET0
```

```

        SETB  ES
        SETB  EA

SOLO:   CALL  PRINT_LCD

        DB   '      ',00
        MOV  A,#LINE2
        MOV  B,#5
        CALL DTSEC
        MOV  CNT,#01
        CALL INIT_AD
;*****MAIN*****
MAIN:
        MOV  B,#2
        CALL DTSEC
        CALL PRINTSTR
        CALL RX_BYE
        JMP  MAIN
;*****A/D*****
INIT_AD:
        CLR  CS
        CLR  SDI
        CLR  SDO
        MOV  R7,#08

INIT1:  MOV  R6,#FF
INIT2:  SETB SCLK
        CLR  SCLK
        DJNZ R6,INIT2
        DJNZ R7,INIT1
        MOV  R7,#08

INIT3:  SETB SCLK

```

```

CLR SCLK
DJNZ R7,INIT3
;*****2*****
SETB SDI
MOV R7,#FF
INIT4: SETB SCLK
CLR SCLK
DJNZ R7,INIT4
;*****3*****
CLR SDI
SETB SCLK
CLR SCLK
;*****CLK*****
XX1: CALL RD_AD
JNB DF,XX1
CALL PRINTSTR
DB 'OK',00
;*****END MAIN*****
LCD_AD: CALL RD_AD
MOV A,#2
CALL GOTO_LCD
MOV B,HI_BUF
MOV DPH,MID_BUF
MOV DPL,LOW_BUF
CALL HTOD
MOV A,R1
CALL LCD_2ASC
MOV A,R2
CALL LCD_2ASC
MOV A,R3

```

```

CALL LCD_2ASC
MOV A,R4
CALL LCD_2ASC
RET
;*****VALUE AD*****
AD_TX_BYTE MOV B,#08
CLR SCLK
AD_TX1: RLC A
MOV SDI,C
SETB SCLK
CLR SCLK
DJNZ B,AD_TX1
CLR SDI
RET
;*****
TOGGLE_SDO: MOV B,#08
CLR SDI
JB SDO
TOGGLE: SETB SCLK
CLK SCLK
DJNZ B,TOGGLE
RET
;*****
SCAN_RX: MOV R6,#5
CLR C
SCAN_RX0: MOV R7,#8
MOV A,@R0
SCAN_RX1: RRC A
NOP
RET

```

```

;*****
HTOSEG:  PUSH  ACC

        SWAP  A

        ANL   A,#0F

        MOV   DPTR,#SEG1

        MOVC  A,@A+DPTR

        CPL   A

        MOV   BUF_959,A

        POP   ACC

        RET

;*****
SEG: DB 3FH,06H,5BH,4FH,66H,6DH,7DH,07H

      DB 7FH,6FH,77H,7CH,58H,DEH,F9H,F1H

      DB 00H,01H,02H,04H,08H,10H,40H,F7H

;*****
LCD_DEC:  PUSH  ACC

         SWAP  A

         ANL   A,#0FH

         ORL   A,#30H

         CALL  WR_LCD

         POP   ACC

         ANL   A,#0FH

         ORL   A,#30H

         CALL  WR_LCD

         RET

;*****
;*****
;Function Add32 IN:(R7:R6:R5:R4)+(R3:R2:R1:R0) OUT=(R7:R6:R5:R4)
;If it has CARRY = Overflow
;*****/

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Add32:

MOV A,R4

ADD A,R0

MOV R4,A

MOV A,R5

ADDC A,R1

MOV R5,A

MOV A,R6

ADDC A,R2

MOV R6,A

MOV A,R7

ADDC A,R3

MOV R7,A

RET

;Function Sub32 IN:(R7:R6:R5:R4)-(R3:R2:R1:R0) OUT=(R7:R6:R5:R4)

;If it has CARRY = (R7:R6:R5:R4) < (R3:R2:R1:R0)

*****/

Sub32:

CLR C

MOV A,R4

SUBB A,R0

MOV R4,A

MOV A,R5

SUBB A,R1

MOV R5,A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV  A,R6
SUBB A,R2
MOV  R6,A

MOV  A,R7
SUBB A,R3
MOV  R7,A

RET

;*****
;Function H2D2 IN:R3, OUT:R4  REGS. USED:ACC,R7
;*****/
H2D2:
MOV  R4,#0    ;RESULT
MOV  R7,#8    ;BitCnt
H2D2Lp:
MOV  A,R3    ;
RLC  A      ;
MOV  R3,A    ;Hex
MOV  A,R4    ;
ADDC A,ACC   ;
DA   A      ;
MOV  R4,A    ;
DJNZ R7,H2D2Lp
RET

;*****
;Function H2D3 IN:R3, OUT:R5:R4  REGS. USED:ACC,R7
;*****/
H2D3:
CLR  A

```

```

MOV R5,A      ;
MOV R4,A      ;RESULT=0
MOV R7,#8     ;Hex BitCnt
H2D3Lp:
MOV A,R3      ;
RLC A         ;
MOV R3,A      ;
MOV A,R4      ;
ADDC A,ACC    ;
DA A         ;
MOV R4,A      ;

MOV A,R5
ADDC A,ACC
DA A
MOV R5,A      ;

DJNZ R7,H2D3Lp
RET

;*****
;Function H3D3 IN:R3:R2, OUT:R5:R4 REGS. USED:ACC,R7
;*****/

H3D3:
CLR A
MOV R5,A      ;
MOV R4,A      ;RESULT=0
MOV R7,#8     ;Hex BitCnt
H3D3Lp:
MOV A,R2      ;
RLC A         ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV R2,A      ;
MOV A,R3      ;
RLC A         ;
MOV R3,A      ;

MOV A,R4      ;
ADDC A,ACC    ;
DA A         ;
MOV R4,A      ;

MOV A,R5
ADDC A,ACC
DA A
MOV R5,A      ;
DJNZ R7,H3D3Lp
RET

;*****
;Function D2H2 IN:R4, OUT:R3 REGS. USED:ACC,R7
;*****/
D2H2:
    MOV R7,#8      ;BitCnt
D2H2Lp:
    CLR C
    MOV A,R4
    RRC A
    PUSH PSW
    CALL D2H
    MOV R4,A      ;
    POP PSW

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV  A,R3
RRC  A
MOV  R3,A
DJNZ R7,D2H2Lp
RET

;*****
;Function D3H2 IN:R5:R4(<=255), OUT:R3  REGS. USED:ACC,R7
;*****/
D3H2:
    MOV  R7,#8      ;BitCnt
D3H2Lp:
    CLR  C
    MOV  A,R5
    RRC  A
    PUSH PSW
    CALL D2H
    MOV  R5,A      ;
    POP  PSW
    MOV  A,R4
    RRC  A
    PUSH PSW
    CALL D2H
    MOV  R4,A      ;
    POP  PSW
    MOV  A,R3
    RRC  A
    MOV  R3,A
    DJNZ R7,D3H2Lp
    RET

;*****

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D2H:

JNB ACC.7,D2H1

CLR C

SUBB A,#30H

D2H1:

JNB ACC.3,D2H_2

CLR C

SUBB A,#3

D2H_2:

RET

;*****

;Function DivC IN:R5:R4,OUT:R4 REGS USED:ACC,R7,R6

R6:Remainder,R5:R4=Quotient

;*****/

DivC: ;IN=R3:R2/Const ;R6 = Remainder,R2=Quotient

CJNE R4,#LOW 6040,\$+3

MOV A,R5

SUBB A,#HIGH 6040

JNC DivCE

MOV R7,#16 ;BitCnt

MOV R6,#0 ;Clear Temp

CLR C

DivCLp:

MOV A,R4

RLC A

MOV R4,A

MOV A,R5

RLC A

MOV R5,A

```

MOV  A,R6
RLC  A
MOV  R6,A

CJNE R6,#60,$+3
JC   DivC1
MOV  A,R6
SUBB A,#60
MOV  R6,A
DivC1:
CPL  C
DJNZ R7,DivCLp
MOV  A,R4
RLC  A
MOV  R4,A
SETB C
DivCE:
RET
,*****
*****
;Function Mul16x8:IN (R5:R4)*R3 REGS USED:ACC,R7 RESULT=R6:R5:R4
,*****
*****
Mul16X8:
MOV  R6,#0   ;Clear Result Most Sign.Byte
MOV  R7,#16  ;Bit Of Multiplicand
Mul16X8Lp:
MOV  A,R4    ;
CLR  C
JNB  ACC.0,Mul16X8A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV  A,R3
ADD  A,R6  ;
MOV  R6,A
Mul16X8A:
MOV  A,R6
RRC  A
MOV  R6,A
MOV  A,R5
RRC  A
MOV  R5,A
MOV  A,R4
RRC  A
MOV  R4,A
DJNZ R7,Mul16X8Lp
RET
;*****
;*****
;Function Mul16x16:IN (R5:R4)*R3:R2 REGS USED:ACC,R1
RESULT=R7:R6:R5:R4
;*****
;*****
Mul16X16:
MOV  R6,#0  ;Clear Result Most Sign.Byte
MOV  R1,#16  ;Bit Of Multiplcand
Mul16X16Lp:
MOV  A,R4  ;
CLR  C
JNB  ACC.0,Mul16X16A
MOV  A,R2
ADD  A,R6  ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV R6,A
MOV A,R3
ADDC A,R7
MOV R7,A

```

Mul16X16A:

```

MOV A,R7
RRC A
MOV R7,A
MOV A,R6
RRC A
MOV R6,A
MOV A,R5
RRC A
MOV R5,A
MOV A,R4
RRC A
MOV R4,A
DJNZ R1,Mul16X16Lp
RET

```

;Function Div16D8:IN (R5:R4)/R3 REGS USED:ACC,R6

R6:Remainder,R5:R4=Quotient

Div16D8:

```

MOV R7,#16+1 ;BitCnt
MOV R6,#0 ;Clear Temp
CLR C

```

Div16D8Lp:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV  A,R4
RLC  A
MOV  R4,A
MOV  A,R5
RLC  A
MOV  R5,A      ;Rotate Dividend
DJNZ R7,Div16D8A
RET

Div16D8A:
MOV  A,R6
RLC  A
MOV  R6,A      ;Rotate Temp
MOV  F0,C
CLR  C
MOV  A,R6
SUBB A,R3      ;
MOV  R6,A
JNC  Div16D8B  ;If Temp => Divider
JB   F0,Div16D8Lp
MOV  A,R6
ADD  A,R3      ;
MOV  R6,A

Div16D8B:
CPL  C
SJMP Div16D8Lp

;*****
;*****
;Function Div24D8:IN (R5:R4:R3)/R2 REGS USED:ACC,R1
R6:Remainder,R5:R4:R3=Quotient
;*****

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Div24D8:

```
MOV R1,#24+1 ;BitCnt
CLR A
MOV R6,A ;Clear Temp
CLR C
```

Div24D8Lp:

```
MOV A,R3
RLC A
MOV R3,A
MOV A,R4
RLC A
MOV R4,A
MOV A,R5
RLC A
MOV R5,A ;Rotate Dividend
DJNZ R1,Div24D8A
RET
```

Div24D8A:

```
MOV A,R6
RLC A
MOV R6,A ;Rotate Temp
MOV F0,C
CLR C
MOV A,R6
SUBB A,R2 ;
MOV R6,A
JNC Div24D8B ;If Temp => Divider
JB F0,Div24D8Lp
MOV A,R6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ADD  A,R2      ;
MOV  R6,A
Div24D8B:
CPL  C
SJMP Div24D8Lp
;*****
;*****
;Function Div24D16:IN (R5:R4:R3)/R2,R1 REGS USED:ACC,R0
R7:R6:Remainder,R5:R4:R3=Quotient
;*****
;*****
Div24D16:
MOV  R0,#24+1 ;BitCnt
MOV  R6,#0    ;Clear Temp
CLR  C
Div24D16Lp:
MOV  A,R3
RLC  A
MOV  R3,A
MOV  A,R4
RLC  A
MOV  R4,A
MOV  A,R5
RLC  A
MOV  R5,A    ;Rotate Dividend
DJNZ R0,Div24D16A
RET
Div24D16A:
MOV  A,R6
RLC  A

```

```

MOV R6,A          ;Rotate Temp
MOV A,R7
RLC A
MOV R7,A          ;Rotate Temp
MOV F0,C
CLR C
MOV A,R6
SUBB A,R1
MOV R6,A
MOV A,R7
SUBB A,R2
MOV R7,A
JNC Div24D16B     ;If Temp => Divider
JB F0,Div24D16Lp
MOV A,R6
ADD A,R1          ;
MOV R6,A
MOV A,R7
ADDC A,R2
MOV R7,A
Div24D16B:
CPL C
SJMP Div24D16Lp
;*****
*****
;Function Div32D16:IN (R5:R4:R3:R2)/DPH:DPL REGS USED:ACC,R1
R7:R6:Remainder,R5:R4:R3:R2=Quotient
;*****
*****
Div32D16:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV R1,#32+1      ;BitCnt
MOV R6,#0         ;Clear Temp
CLR C
Div32D16Lp:
MOV A,R2
RLC A
MOV R2,A
MOV A,R3
RLC A
MOV R3,A
MOV A,R4
RLC A
MOV R4,A
MOV A,R5
RLC A
MOV R5,A         ;Rotate Dividend
DJNZ R1,Div32D16A
RET
Div32D16A:
MOV A,R6
RLC A
MOV R6,A         ;Rotate Temp
MOV A,R7
RLC A
MOV R7,A         ;Rotate Temp
MOV F0,C
CLR C
MOV A,R6
SUBB A,DPL
MOV R6,A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV  A,R7
SUBB  A,DPH
MOV  R7,A
JNC  Div32D16B      ;If Temp => Divider
JB   F0,Div32D16Lp
MOV  A,R6
ADD  A,DPL      ;
MOV  R6,A
MOV  A,R7
ADDC A,DPH
MOV  R7,A
Div32D16B:
CPL  C
SJMP Div32D16Lp
;*****
;*****
; ***** HTOD 24 BIT SUB *****
; HEX TO DECIMAL
; IN = B,DPTR
; OUT = R1,R2,R3,R4
; REG = A,R0,R1,R2,R3,R4,R5,R7,DPTR

HTOD: CLR  A      ;CLEAR OUTPUT
      MOV  R1,A
      MOV  R2,A
      MOV  R3,A
      MOV  R4,A

      MOV  R7,#24  ;SHIFT 16 BIT
HTOD1: MOV  A,DPL

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RLC A
MOV DPL,A
MOV A,DPH
RLC A
MOV DPH,A
MOV A,B
RLC A
MOV B,A

MOV R5,#4 ;ADD DECIMAL
MOV R0,#4 ;INDEX TO R4
HTOD2: MOV A,@R0
ADDC A,ACC
DA A
MOV @R0,A
DEC R0
DJNZ R5,HTOD2
DJNZ R7,HTOD1
RET

; ***** DTOH SUB *****
; DECIMAL TO HEX
; IN = R1,R2,R3
; OUT = DPTR
; REG = A,R0,R1,R2,R3,R4,R5,DPTR

DTOH: MOV R4,#16
DTOH1: MOV R5,#3 ;SHIFT & SUB
MOV R0,#1 ;INDEX TO R1
CLR C

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DTH02: MOV  A,@R0

```

```

    RRC  A

```

```

    PUSH PSW      ;-[

```

```

    JNB  ACC.7,DTH03

```

```

    CLR  C

```

```

    SUBB A,#30H

```

```

DTH03: JNB  ACC.3,DTH04

```

```

    CLR  C

```

```

    SUBB A,#03H

```

```

DTH04: MOV  @R0,A

```

```

    INC  R0

```

```

    POP  PSW      ;:]

```

```

    DJNZ R5,DTH02

```

```

    MOV  A,DPH

```

```

    RRC  A

```

```

    MOV  DPH,A

```

```

    MOV  A,DPL

```

```

    RRC  A

```

```

    MOV  DPL,A

```

```

    DJNZ R4,DTH01

```

```

    RET

```

```

;***** HEX TO ASCII CONVERT SUB *****

```

```

;   IN  = A

```

```

;   OUT = R2,R3

```

```

HTOA:   PUSH  ACC

```

```

        SWAP  A

```

```

        LCALL HTOAS

```

```

        MOV  R2,A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    POP    ACC
    LCALL HTOAS
    MOV    R3,A
    RET

HTOAS:  ANL  A,#0FH
        CJNE A,#0AH,XHTOAS
        JNC  HTOAS1
        ORL  A,#30H

XHTOAS:  RET
HTOAS1:  SUBB A,#9
        ORL  A,#40H
        RET

;***** ATOH SUB *****
; ASCII TO HEX CONVERT
; IN = R2,R3  30H,41H
; OUT = A  0AH
ATOH:   MOV    A,R2
        CALL  ATOHS
        SWAP  A
        MOV   R2,A
        MOV   A,R3
        LCALL ATOHS
        ORL   A,R2
        RET

ATOHS:  CJNE  A,#'A',$+3
        JC   ATOHS1
        ADD  A,#9
ATOHS1: ANL   A,#0FH
        RET

;***** DTSEC SUB *****

```

```

; DELAY MILISECOND
; IN = B
; REG = A,B,DPTR
DTMSEC: MOV DPTR,#65536-152
MD1: INC DPTR
      MOV A,DPH
      ORL A,DPL
      JNZ MD1
      NOP
      DJNZ B,DTMSEC
      RET

; ***** DTSEC SUB *****
; DELAY 1/10 SECOND
; IN = B
; REG = B,R2,R3
DTSEC: MOV R2,#179
DTSEC1: MOV R3,#0
        DJNZ R3,$
        NOP
        NOP
        DJNZ R2,DTSEC1
        DJNZ B,DTSEC
        RET

; ***** DTMSEC SUB *****
; DELAY 1X10 MICRO SECOND
; IN = B
; REG = B
DTMCSEC: NOP

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DJNZ B,DTMCSEC

RET

;*****
;* Send 2 byte of ASCII *
;* Input : ACC (HEX) *
;* Example : A = A3H *
;*      : send 41H,33H *
;*****
;

TX_2ASC: LCALL HEX_2ASC
        LCALL TX_BYTE
        MOV  A,B
        LCALL TX_BYTE
        RET

;*****
;* Convert HEX to ASCII *
;* Input  : A *
;* Output : A (high-byte) *
;*      : B (low-byte) *
;*****
;

HEX_2ASC: PUSH ACC
        LCALL HEX_TO_ASC
        MOV  B,A      ; B = Lo byte
        POP  ACC
        SWAP A
        LCALL HEX_TO_ASC
        RET

```

```

;*****
;* Convert hex (0-F) to ASCII *
;* Input  : ACC (HEX code) *
;* Output : ACC (ASCII code) *
;*****
;
HEX_TO_ASC: PUSH PSW
            CLR CY
            ANL A,#00001111B
            CJNE A,#09H,HTOA1
            SETB CY
HTOA1:     JNB CY,HTOA2    ;> 0AH
            ORL A,#30H     ;< 0AH
            SJMP END_HTOA
HTOA2:     SUBB A,#09H    ;> 0AH
            ORL A,#40H
END_HTOA:  POP PSW
            RET
;***** CONV 1BYTE(BIN) TO 3BYTE(ASCII) *****
;TX_3ASC:  lcall step    ;1byte to 3digit(0-255)
            mov a,r3
            lcall HEX_TO_ASC ;decimal value to Ascii(0-9)
            lcall TX_BYTE
            mov a,r2
            lcall HEX_TO_ASC
            lcall TX_BYTE
            mov a,r1
            lcall HEX_TO_ASC
            lcall TX_BYTE
            ret

```

```

;*****
;* Send 1 byte BCD to ASCII form *
;* Input  : ACC = Binary      *
;* Output : BCD To rs232 ch  *
;* Register: ACC.,B.        *
;*****
;
SEND_BCD: CLR  TI
          MOV  B,A
          SWAP A
          ACALL SBCD
          MOV  A,B
SBCD:    ANL  A,#0FH
          CJNE A,#0AH,SBCD2
SBCD2:   JC   SBCD3
          ADD  A,#55
          SJMP SBCD4
SBCD3:   ORL  A,#30H
SBCD4:   CLR  ES
          MOV  SBUF,A
          JNB  TI,$
          CLR  TI
          SETB ES
          RET

;*****
;* PRINTSTR = SEND STRING      *
;* FUNTION  = Send string at memory - *
;*          location after instruction*
;*          lcall PRINT instruction *

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;*      end of string must be 00  *
;*****
;
;Note that the end of string table must be 00
;NOTE : when used this sub routine must not enable ES
;
PRINTSTR: POP  DPH    ; Get Pointer(stack)
          POP  DPL
          PUSH IE
          PUSH ACC
          CLR  ES
STSD0:   CLR  A
          MOVC A,@A+DPTR
          JZ  STSD2
STSD1:   CLR  TI
          MOV  SBUF,A
          JNB TI,$    ;Wait to send
          CLR  TI
          INC  DPTR
          SJMP STSD0
STSD2:   POP  ACC
          POP  IE
          INC  DPTR
          PUSH DPL
          PUSH DPH
          RET
;*****
;* RX_BYTE = Receive byte      *
;* FUNCTION : Receive 1 byte from *

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

,*      serial port.      *
,* Output : Register ACC.      *
,******
;
RX_BYTE: PUSH IE
        CLR ES
        CLR RI
        JNB RI,$
RX_BT1: MOV  A,SBUF
        CLR RI
        POP IE
        RET
,******
,* TX_BYTE = 1 BYTE SEND      *
,* FUNCTION = Send data in A to serial*
,*      port.      *
,* INPUT  = Data in A to transmit *
,* OUTPUT = Serial port      *
,******
;
TX_BYTE: PUSH IE
        CLR ES
        CLR TI
        MOV  SBUF,A
        JNB TI,$
        CLR TI
        POP IE
        RET

```

```

$INCLUDE "XDELAY.ASM"

$INCLUDE "XRS232.ASM"

$INCLUDE "LCD_164.ASM"

$INCLUDE "XCONV_1.ASM"

END

```

โปรแกรมการทำงาน T.C. Calibration

```

;*****
;* Temp Calibrator Program *
;* uP : AT89C51 *
;* Memory : *
;* Speaker : *
;* WD : *
;* Key SW : 12 KeyTel *
;* Display : LCD *
;* Com Port : *

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;* BY MR. Winai P. *
;* LAST UPDATE Nov 17'00 *
;* TIME 13:35 *
;*****
;**** A/D ****
CS BIT P3.4 ;pin 18
CLK BIT P3.5 ;pin 17
DO BIT P3.6 ;pin 14
DI BIT P3.7 ;pin 11

AC_LED BIT P1.0
;***** LCD *****
RS_LCD BIT P2.5
CS_LCD BIT P2.4
ORG 21H
Flag: DS 1
cnt: ds 01
;
COM_REQ BIT Flag.1
kb_Req bit Flag.2
kb_full bit Flag.3
t_out bit Flag.4

;***** TIMER0 *****
CLK_208MC: DS 1H
CLK_5MS: DS 1H
Ms_Clk: ds 2
Sec_Clk: ds 1
Sec_x1: ds 1
TIME_OUT: DS 1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

COMM_BUF:  DS  1
;***** cs5526 *****
COMMAND:   DS  1
kb_buf:    ds  3
deg_inx:   ds  2
deg:       ds  2
kb_hex:    ds  2
ad_buf:    ds  2
Err_Val:   ds  2
key_buf:   ds  1
;***** ADC0832 *****
MSEC:      DS  1
DVALUE:    DS  1
CH         EQU  20H
STACK      EQU  $

LF         EQU  0AH  ;LINE FEED
CR         EQU  0DH  ;RETURN
LINE1      EQU  00
LINE2      EQU  0C0H
LINE3      EQU  0d0H
LINE4      EQU  90H

;-----
ORG 0000H
SJMP P_RESET
ORG 0BH
LJMP TIMER0_INT
ORG 23H
LJMP SERIAL_INT

P_RESET:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;***** SET SERIAL PARAMETER *****

MOV   PCON,#80H
MOV   TMOD,#22H
MOV   TH1,#0FAH
MOV   TL1,#0FAH
SETB  TCON.6      ;stop/start timer
MOV   SCON,#052H  ;8 bit UART

;-----
;***** SET INT PARAMEEER *****

MOV   SP,#(STACK-1) ; Initial Stack
MOV   TH0,#256-192 ;10H
MOV   TL0,#256-192 ;10H
STITLE: CALL  INIT_LCD
SETB  TR0
SETB  ET0
SETB  ES
SETB  EA

;***** INIT & TITLE DISPLAY *****
aaa:  CALL  PRINT_LCD
DB    ' T.C Calibration',00
MOV   A,#LINE2
MOV   B,#5
CALL  DTSEC
MOV   DPTR,#De_tab ;TEST

inc   dptr
inc   dptr
inc   dptr
inc   dptr

MOV   A,#00
MOVC  A,@A+DPTR

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV    A,#01
MOV    A,@A+DPTR
;
;*****
;***** MAIN
;*****
;
;
MOV    B,#2
CALL   DTSEC
mov    a,cnt
;*****
call   w_dep_key
call   del_key
call   w_prs_key
;
MAIN:  MOV    CH,#0
L123:  CALL   AD0832
CALL   AD0832
CALL   AD0832
CALL   AD0832
MOV    A,DVALUE
MOV    AD_BUF,A
mov    ad_buf+1,#00H ;!!!!
; MOV  P1,DVALUE
; SJMP L123
;*****
mov    b,#10
call   dtsec
call   get_3d
call   kb_deg

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;call De_tab ;/////////
call show_deg
call show_DP
call show_Err
call w_dep_key
call del_key
call w_prs_key
;
JMP main
;
get_3d: mov a,#line1
call goto_lcd
call print_lcd
db 'Set: 0000 c ',00
call clear_l2
mov a,#line1+7
call goto_lcd
call w_prs_key
mov a,key_buf
call wr_lcd
mov a,key_buf
call atohs
mov kb_buf,a
call w_dep_key
call del_key
call w_prs_key
mov a,key_buf
call wr_lcd
mov a,key_buf
call atohs

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov    kb_buf+1,a
call   w_dep_key
call   del_key
call   w_prs_key
mov    a,key_buf
call   wr_lcd
mov    a,key_buf
call   atohs
mov    kb_buf+2,a
;*****
mov    a,kb_buf
mov    a,kb_buf+1
mov    a,kb_buf+2
call   del_key
ret
kb_deg: mov    a,kb_buf+1
mov    b,#10
mul    ab
add    a,kb_buf+2
mov    deg_inx,a
mov    r5,#00
mov    r4,kb_buf
mov    r3,#100
call   mul16x8
mov    r6,#00
mov    r7,#00
mov    r0,deg_inx
mov    r1,#00
mov    r2,#00
mov    r3,#00

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    call    add32
    mov     kb_hex,r4
    mov     kb_hex+1,r5
;***** conv
    mov     r3,#2
    call    mul16x8
    mov     deg_inx,r4
    mov     deg_inx+1,r5

    ret

show_kb_hex: mov     a,#line2
             call    goto_lcd
             call    print_lcd
             db     'kb_hex=',00
             mov     dpl,kb_hex
             mov     dph,kb_hex+1
             mov     b,#00
             call    htod
             mov     a,r3
             call    lcd_2asc
             mov     a,r4
             call    lcd_2asc
             ret

show_inx:   mov     a,#line4
             call    goto_lcd
             call    print_lcd
             db     'deg_inx',00
             mov     dpl,deg_inx
             mov     dph,deg_inx+1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov    b,#00
call   htod
mov    a,r3
call   lcd_2asc
mov    a,r4
call   lcd_2asc
ret

show_deg:  mov    dptr,#De_tab
           mov    r4,dpl
           mov    r5,dph
           mov    r6,#00
           mov    r7,#00
           mov    r0,deg_inx
           mov    r1,deg_inx+1
           mov    r2,#00
           mov    r3,#00
           call   add32
           mov    dpl,r4
           mov    dph,r5
           mov    a,#00
           MOVC   A,@A+DPTR
           mov    deg,a
           mov    a,#01
           MOVC   A,@A+DPTR
           mov    deg+1,a
           mov    a,#line1
           call   goto_lcd
           call   print_lcd
           db     'deg=',00
           mov    dpl,deg

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov    dph,deg+1
mov    b,#00
call   htod
mov    a,deg ;r3
call   lcd_2asc
mov    a,deg+1 ;r4
call   lcd_2asc
ret

show_dp:  mov    a,#line1
call    goto_lcd
call    print_lcd
db     ' Std=',00
mov    dpl,deg+1
mov    dph,deg
mov    b,#00
call   htod
mov    a,r2
CALL   HEX_2ASC
MOV    A,B
CALL   WR_LCD
mov    a,r3
LCALL  HEX_2ASC
LCALL  WR_LCD
mov    a,#'!'
call   wr_lcd
MOV    A,B
LCALL  WR_LCD
mov    a,r4
call   lcd_2asc

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    call  print_lcd
    db    'mV ',00
    ret
NE_:   AJMP  NE_1
show_Err:  mov  a,#line2
        call  goto_lcd
        MOV   A,DEG
        MOV   B,AD_BUF
        CLR   CY
        SUBB  A,B
        JC    NE_
        call  print_lcd
        db    'Err= ',00
        mov   r5,deg
        mov   r4,deg+1
        mov   r6,#00
        mov   r7,#00
        mov   r0,ad_buf
        mov   r1,ad_buf+1
        mov   r2,#00
        mov   r3,#00
        call  sub32
        mov   Err_Val,r4
        mov   Err_Val+1,r5
        mov   dpl,r4
        mov   dph,r5
        mov   b,#00
        call  htod
        mov   a,r2
        CALL  HEX_2ASC

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV A,B
CALL WR_LCD
mov a,r3
LCALL HEX_2ASC
LCALL WR_LCD
mov a,'#.'
call wr_lcd
MOV A,B
LCALL WR_LCD
mov a,r4
call lcd_2asc
call print_lcd
db 'mV',00
ret
NE_1: call print_lcd
db 'Err= ',00
mov r5,AD_BUF
mov r4,AD_BUF+1
mov r6,#00
mov r7,#00
mov r0,DEG
mov r1,DEG+1
mov r2,#00
mov r3,#00
call sub32
mov Err_Val,r4
mov Err_Val+1,r5
mov dpl,r4
mov dph,r5
mov b,#00

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

call  htod
mov   a,r2
CALL  HEX_2ASC
MOV   A,B
CALL  WR_LCD
mov   a,r3
LCALL HEX_2ASC
LCALL WR_LCD
mov   a,#'.'
call  wr_lcd
MOV   A,B
LCALL WR_LCD
mov   a,r4
call  lcd_2asc
call  print_lcd
db    'mV ',00
ret

del_key:  mov   b,#2
          call  dtsec
          ret

;***** Scan 1 Line *****
;      in    r0 = Start Buf
scan_rx:  mov   r6,#5
          clr   c
scan_rx_0:  mov   r7,#8    ;80 bit
          mov   a,@r0
;*****

TIMER0_INT:  PUSH  ACC
             PUSH  PSW

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

INC   CLK_208MC
MOV   A,CLK_208MC
CJNE  A,#24,Q_TIMER0_INT
MOV   CLK_208MC,#00
INC   CLK_5MS
MOV   A,CLK_5MS
CJNE  A,#200,Q_TIMER0_INT
MOV   CLK_5MS,#00
JMP   SEC_INC

Q_TIMER0_INT:
POP   PSW
POP   ACC
RETI

Sec_INC:
DJNZ  TIME_OUT,SEC_INC2
SETB  T_OUT

SEC_INC2: INC   Sec_x1
INC   SEC_CLK
MOV   A,SEC_CLK

CPL   AC_LED
CJNE  A,#60,Q_timer0_Int

Reduce_Min: mov   Sec_Clk,#00
JMP   Q_TIMER0_INT

;***** SERIAL INT *****
SERIAL_INT: JB   TI,XMIT
JB   RI,XREC
SJMP  Q_SER_INT

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

XMIT:    CLR  TI
Q_SER_INT:  RETI
XREC:    CLR  RI
         PUSH ACC
         PUSH PSW
         MOV  A,SBUF
         MOV  COMM_BUF,A
         SETB COM_REQ
Q_INT:   POP  PSW
         POP  ACC
         RETI
;*****
TEST:    DW   1234H,5678H
;***** PRINT 2ASC TO LCD *****
LCD_2ASC:  LCALL HEX_2ASC
          LCALL WR_LCD
          MOV  A,B
          LCALL WR_LCD
          RET
LCD_DEC:   PUSH ACC
          SWAP A
          ANL  A,#0FH
          ORL  A,#30H
          CALL WR_LCD
          POP  ACC
          ANL  A,#0FH
          ORL  A,#30H
          CALL WR_LCD
          RET

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
##### SUB SCAN KEYBOARD #####
```

```
SCANKEY: JB KB_FULL,NO_KEY
```

```
MOV p0,#0FFH
```

```
MOV R0,#11101111B
```

```
MOV KEY_BUF,#00
```

```
MOV R1,#4 ;4 ROW
```

```
MOV R2,#3 ;3 COLUMN
```

```
SCAN: MOV A,R0
```

```
MOV P0,A
```

```
NOP
```

```
MOV A,P0
```

```
SCAN1: RRC A
```

```
INC KEY_BUF
```

```
JNC PRS_KEY
```

```
DJNZ R1,SCAN1
```

```
MOV A,R0
```

```
RL A
```

```
MOV R0,A
```

```
MOV R1,#4
```

```
DJNZ R2,SCAN
```

```
NO_KEY: RET
```

```
PRS_KEY: SETB KB_REQ
```

```
SETB KB_FULL
```

```
MOV A,KEY_BUF
```

```
MOV DPTR,#KB_TABLE
```

```
MOVC A,@A+DPTR
```

```
MOV KEY_BUF,A
```

```
RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;***** SCAN KEY & WAIT DEPRESS KEY *****
SCANKEY_W:  CLR  T_OUT
            MOV  TIME_OUT,#20
            CLR  KB_FULL
            CLR  KB_REQ

SCANKEY_W1: CALL  SCANKEY
            MOV  A,TIME_OUT
            JZ   SCAN_T_OUT
            JNB  KB_REQ,SCANKEY_W1
            MOV  A,KEY_BUF
            PUSH ACC
            CALL W_DEP_KEY
            POP  ACC
            MOV  KEY_BUF,A
            RET

SCAN_T_OUT: SETB  T_OUT
            RET

;*****

;***** WAIT DEPRESS KEYBOARD *****
W_DEP_KEY:  CLR  KB_FULL
W_DEP_KEY1: CLR  KB_REQ
            CALL SCANKEY
            MOV  P1,kb_hex
            JB   KB_REQ,W_DEP_KEY1
            RET

;***** WAIT PRESS KEY *****
W_PRS_KEY:  CLR  KB_FULL
            CLR  KB_REQ
W_PRS_KEY1: CALL  SCANKEY

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        JNB  KB_REQ,W_PRS_KEY1
        RET

KB_TABLE:
        DB  '*321#6540987'

DL_KB:  MOV  B,#2
        CALL DTSEC
        RET

clear_l2:  mov  a,#line2
          call goto_lcd
          call print_lcd
          db  ' ',00
          ret

;*****
;Function Add32 IN:(R7:R6:R5:R4)+(R3:R2:R1:R0) OUT=(R7:R6:R5:R4)
;If it has CARRY = Overflow
;*****/

Add32:
        MOV  A,R4
        ADD  A,R0
        MOV  R4,A

        MOV  A,R5
        ADDC A,R1
        MOV  R5,A

        MOV  A,R6
        ADDC A,R2
        MOV  R6,A

```

```

MOV  A,R7
ADDC A,R3
MOV  R7,A
RET

```

```

;*****

```

```

;Function Sub32 IN:(R7:R6:R5:R4)-(R3:R2:R1:R0) OUT=(R7:R6:R5:R4)

```

```

;If it has CARRY = (R7:R6:R5:R4) < (R3:R2:R1:R0)

```

```

;*****/

```

```

Sub32:

```

```

CLR  C
MOV  A,R4
SUBB A,R0
MOV  R4,A

MOV  A,R5
SUBB A,R1
MOV  R5,A

MOV  A,R6
SUBB A,R2
MOV  R6,A

MOV  A,R7
SUBB A,R3
MOV  R7,A
RET

```

```

;*****

```

```

;Function H2D2 IN:R3, OUT:R4 REGS. USED:ACC,R7

```

```

;*****/

```

```

H2D2:

```

```

MOV R4,#0 ;RESULT
MOV R7,#8 ;BitCnt
H2D2Lp:
MOV A,R3 ;
RLC A ;
MOV R3,A ;Hex
MOV A,R4 ;
ADDC A,ACC ;
DA A ;
MOV R4,A ;
DJNZ R7,H2D2Lp
RET
;*****
;Function H2D3 IN:R3, OUT:R5:R4 REGS. USED:ACC,R7
;*****/
H2D3:
CLR A
MOV R5,A ;
MOV R4,A ;RESULT=0
MOV R7,#8 ;Hex BitCnt
H2D3Lp:
MOV A,R3 ;
RLC A ;
MOV R3,A ;
MOV A,R4 ;
ADDC A,ACC ;
DA A ;
MOV R4,A ;
MOV A,R5

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    ADDC A,ACC
    DA A
    MOV R5,A ;

    DJNZ R7,H2D3Lp
    RET

;*****
;Function H3D3 IN:R3:R2, OUT:R5:R4 REGS. USED:ACC,R7
;*****/
H3D3:
    CLR A
    MOV R5,A ;
    MOV R4,A ;RESULT=0
    MOV R7,#8 ;Hex BitCnt
H3D3Lp:
    MOV A,R2 ;
    RLC A ;
    MOV R2,A ;
    MOV A,R3 ;
    RLC A ;
    MOV R3,A ;

    MOV A,R4 ;
    ADDC A,ACC ;
    DA A ;
    MOV R4,A ;

    MOV A,R5
    ADDC A,ACC
    DA A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV R5,A      ;

DJNZ R7,H3D3Lp
RET

;*****
;Function D2H2 IN:R4, OUT:R3 REGS. USED:ACC,R7
;*****/

D2H2:
MOV R7,#8      ;BitCnt
D2H2Lp:
CLR C
MOV A,R4
RRC A
PUSH PSW
CALL D2H
MOV R4,A      ;
POP PSW
MOV A,R3
RRC A
MOV R3,A
DJNZ R7,D2H2Lp
RET

;*****
;Function D3H2 IN:R5:R4(<=255), OUT:R3 REGS. USED:ACC,R7
;*****/

D3H2:
MOV R7,#8      ;BitCnt
D3H2Lp:
CLR C
MOV A,R5

```

```

RRC A
PUSH PSW
CALL D2H
MOV R5,A ;
POP PSW
MOV A,R4
RRC A
PUSH PSW
CALL D2H
MOV R4,A ;
POP PSW
MOV A,R3
RRC A
MOV R3,A
DJNZ R7,D3H2Lp
RET
;*****
D2H:
JNB ACC.7,D2H1
CLR C
SUBB A,#30H
D2H1:
JNB ACC.3,D2H_2
CLR C
SUBB A,#3
D2H_2:
RET
;*****
;Function DivC IN:R5:R4,OUT:R4 REGS USED:ACC,R7,R6
R6:Remainder,R5:R4=Quotient

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;*****/

```

```

DivC: ;IN=R3:R2/Const ;R6 = Remainder,R2=Quotient

```

```

    CJNE R4,#LOW 6040,$+3

```

```

    MOV  A,R5

```

```

    SUBB A,#HIGH 6040

```

```

    JNC  DivCE

```

```

    MOV  R7,#16      ;BitCnt

```

```

    MOV  R6,#0      ;Clear Temp

```

```

    CLR  C

```

```

DivCLp:

```

```

    MOV  A,R4

```

```

    RLC  A

```

```

    MOV  R4,A

```

```

    MOV  A,R5

```

```

    RLC  A

```

```

    MOV  R5,A

```

```

    MOV  A,R6

```

```

    RLC  A

```

```

    MOV  R6,A

```

```

    CJNE R6,#60,$+3

```

```

    JC   DivC1

```

```

    MOV  A,R6

```

```

    SUBB A,#60

```

```

    MOV  R6,A

```

```

DivC1:

```

```

    CPL  C

```

```

    DJNZ R7,DivCLp

```

```

    MOV  A,R4

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RLC  A
MOV  R4,A
SETB C
DivCE:
RET
;*****
;*****
;Function Mul16x8:IN (R5:R4)*R3 REGS USED:ACC,R7 RESULT=R6:R5:R4
;*****
;*****
Mul16X8:
MOV  R6,#0 ;Clear Result Most Sign.Byte
MOV  R7,#16 ;Bit Of Multiplicand
Mul16X8Lp:
MOV  A,R4 ;
CLR  C
JNB  ACC.0,Mul16X8A
MOV  A,R3
ADD  A,R6 ;
MOV  R6,A
Mul16X8A:
MOV  A,R6
RRC  A
MOV  R6,A
MOV  A,R5
RRC  A
MOV  R5,A
MOV  A,R4
RRC  A
MOV  R4,A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DJNZ R7,Mul16X8Lp
RET
;*****
****
;Function Mul16x16:IN (R5:R4)*R3:R2 REGS USED:ACC,R1
RESULT=R7:R6:R5:R4
;*****
****
Mul16X16:
MOV R6,#0 ;Clear Result Most Sign.Byte
MOV R1,#16 ;Bit Of Multiplicand
Mul16X16Lp:
MOV A,R4 ;
CLR C
JNB ACC.0,Mul16X16A
MOV A,R2
ADD A,R6 ;
MOV R6,A
MOV A,R3
ADDC A,R7
MOV R7,A
Mul16X16A:
MOV A,R7
RRC A
MOV R7,A
MOV A,R6
RRC A
MOV R6,A
MOV A,R5
RRC A

```

```

MOV R5,A
MOV A,R4
RRC A
MOV R4,A
DJNZ R1,Mul16X16Lp
RET

;*****
;*****
;Function Div16D8:IN (R5:R4)/R3 REGS USED:ACC,R6
R6:Remainder,R5:R4=Quotient
;*****
;*****
Div16D8:
MOV R7,#16+1 ;BitCnt
MOV R6,#0 ;Clear Temp
CLR C
Div16D8Lp:
MOV A,R4
RLC A
MOV R4,A
MOV A,R5
RLC A
MOV R5,A ;Rotate Dividend
DJNZ R7,Div16D8A
RET
Div16D8A:
MOV A,R6
RLC A
MOV R6,A ;Rotate Temp
MOV F0,C

```

```

CLR C
MOV A,R6
SUBB A,R3 ;
MOV R6,A
JNC Div16D8B ;If Temp => Divider
JB F0,Div16D8Lp
MOV A,R6
ADD A,R3 ;
MOV R6,A
Div16D8B:
CPL C
SJMP Div16D8Lp
;*****
****
;Function Div24D8:IN (R5:R4:R3)/R2 REGS USED:ACC,R1
R6:Remainder,R5:R4:R3=Quotient
;*****
****
Div24D8:
MOV R1,#24+1 ;BitCnt
CLR A
MOV R6,A ;Clear Temp
CLR C
Div24D8Lp:
MOV A,R3
RLC A
MOV R3,A
MOV A,R4
RLC A
MOV R4,A

```

```

MOV  A,R5
RLC  A
MOV  R5,A      ;Rotate Dividend
DJNZ R1,Div24D8A
RET
Div24D8A:
MOV  A,R6
RLC  A
MOV  R6,A      ;Rotate Temp
MOV  F0,C
CLR  C
MOV  A,R6
SUBB A,R2      ;
MOV  R6,A
JNC  Div24D8B  ;If Temp => Divider
JB   F0,Div24D8Lp
MOV  A,R6
ADD  A,R2      ;
MOV  R6,A
Div24D8B:
CPL  C
SJMP Div24D8Lp
;*****
****
;Function Div24D16:IN (R5:R4:R3)/R2,R1 REGS USED:ACC,R0
R7:R6:Remainder,R5:R4:R3=Quotient
;*****
****
Div24D16:
MOV  R0,#24+1 ;BitCnt

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV R6,#0           ;Clear Temp
CLR C
Div24D16Lp:
MOV A,R3
RLC A
MOV R3,A
MOV A,R4
RLC A
MOV R4,A
MOV A,R5
RLC A
MOV R5,A           ;Rotate Dividend
DJNZ R0,Div24D16A
RET
Div24D16A:
MOV A,R6
RLC A
MOV R6,A           ;Rotate Temp
MOV A,R7
RLC A
MOV R7,A           ;Rotate Temp
MOV F0,C
CLR C
MOV A,R6
SUBB A,R1
MOV R6,A
MOV A,R7
SUBB A,R2
MOV R7,A
JNC Div24D16B      ;If Temp => Divider

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JB    F0,Div24D16Lp
MOV   A,R6
ADD   A,R1    ;
MOV   R6,A
MOV   A,R7
ADDC  A,R2
MOV   R7,A

Div24D16B:
CPL   C
SJMP  Div24D16Lp
;*****
;*****
;Function Div32D16:IN (R5:R4:R3:R2)/DPH:DPL REGS USED:ACC,R1
R7:R6:Remainder,R5:R4:R3:R2=Quotient
;*****
;*****
Div32D16:
MOV   R1,#32+1    ;BitCnt
MOV   R6,#0       ;Clear Temp
CLR   C
Div32D16Lp:
MOV   A,R2
RLC   A
MOV   R2,A
MOV   A,R3
RLC   A
MOV   R3,A
MOV   A,R4
RLC   A
MOV   R4,A

```

```

MOV  A,R5
RLC  A
MOV  R5,A      ;Rotate Dividend
DJNZ R1,Div32D16A
RET

```

Div32D16A:

```

MOV  A,R6
RLC  A
MOV  R6,A      ;Rotate Temp
MOV  A,R7
RLC  A
MOV  R7,A      ;Rotate Temp
MOV  F0,C
CLR  C
MOV  A,R6
SUBB A,DPL
MOV  R6,A
MOV  A,R7
SUBB A,DPH
MOV  R7,A
JNC  Div32D16B ;If Temp => Divider
JB   F0,Div32D16Lp
MOV  A,R6
ADD  A,DPL      ;
MOV  R6,A
MOV  A,R7
ADDC A,DPH
MOV  R7,A

```

Div32D16B:

```
CPL  C
```

```

SJMP Div32D16Lp
ret
;*****
****
;***** HTOD 24 BIT SUB *****
; HEX TO DECIMAL
; IN = B,DPTR
; OUT = R1,R2,R3,R4
; REG = A,R0,R1,R2,R3,R4,R5,R7,DPTR

HTOD: CLR A ;CLEAR OUTPUT
MOV R1,A
MOV R2,A
MOV R3,A
MOV R4,A

MOV R7,#24 ;SHIFT 16 BIT
HTOD1: MOV A,DPL
RLC A
MOV DPL,A
MOV A,DPH
RLC A
MOV DPH,A
MOV A,B
RLC A
MOV B,A

MOV R5,#4 ;ADD DECIMAL
MOV R0,#4 ;INDEX TO R4
HTOD2: MOV A,@R0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ADDC A,ACC
DA A
MOV @R0,A
DEC R0
DJNZ R5,HTOD2
DJNZ R7,HTOD1
RET

```

```

; ***** DTOH SUB *****
; DECIMAL TO HEX
; IN = R1,R2,R3
; OUT = DPTR
; REG = A,R0,R1,R2,R3,R4,R5,DPTR

DTOH: MOV R4,#16
DTOH1: MOV R5,#3 ;SHIFT & SUB
MOV R0,#1 ;INDEX TO R1
CLR C
DTOH2: MOV A,@R0
RRC A
PUSH PSW ;-[
JNB ACC.7,DTOH3
CLR C
SUBB A,#30H
DTOH3: JNB ACC.3,DTOH4
CLR C
SUBB A,#03H
DTOH4: MOV @R0,A
INC R0
POP PSW ;-]

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DJNZ R5,DTOH2

MOV A,DPH
RRC A
MOV DPH,A
MOV A,DPL
RRC A
MOV DPL,A
DJNZ R4,DTOH1
RET

;***** HEX TO ASCII CONVERT SUB *****
; IN = A
; OUT = R2,R3
HTOA:  PUSH ACC
        SWAP A
        LCALL HTOAS
        MOV R2,A
        POP ACC
        LCALL HTOAS
        MOV R3,A
        RET

HTOAS:  ANL A,#0FH
        CJNE A,#0AH,XHTOAS
        JNC HTOAS1
        ORL A,#30H
XHTOAS:  RET
HTOAS1:  SUBB A,#9
        ORL A,#40H
        RET

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;***** ATOH SUB *****
; ASCII TO HEX CONVERT
; IN = R2,R3 30H,41H
; OUT = A 0AH
ATOH:  MOV  A,R2
      CALL ATOHS
      SWAP A
      MOV  R2,A
      MOV  A,R3
      LCALL ATOHS
      ORL  A,R2
      RET
ATOHS: CJNE A,#'A',$+3
      JC  ATOHS1
      ADD A,#9
ATOHS1: ANL  A,#0FH
      RET
;***** DTSEC SUB *****
; DELAY MILISECOND
; IN = B
; REG = A,B,DPTR
DTMSEC: MOV  DPTR,#65536-152
MD1:  INC  DPTR
      MOV  A,DPH
      ORL  A,DPL
      JNZ  MD1
      NOP
      DJNZ B,DTMSEC
      RET

```

```

;***** DTSEC SUB *****
; DELAY 1/10 SECOND
; IN = B
; REG = B,R2,R3
DTSEC:  MOV R2,#179
DTSEC1: MOV R3,#0
        DJNZ R3,$
        NOP
        NOP
        DJNZ R2,DTSEC1
        DJNZ B,DTSEC
        RET

;***** DTMSEC SUB *****
; DELAY 1X10 MICRO SECOND
; IN = B
; REG = B
DTMCSEC: NOP
        DJNZ B,DTMCSEC
        RET

;*****
;* Send 2 byte of ASCII *
;* Input : ACC (HEX) *
;* Example : A = A3H *
;*      : send 41H,33H *
;*****
;
TX_2ASC: LCALL HEX_2ASC
        LCALL TX_BYTE
        MOV A,B

```

```

    LCALL TX_BYTE
    RET

;*****
;* Convert HEX to ASCII *
;* Input  : A      *
;* Output : A (high-byte) *
;*       : B (low-byte) *
;*****
;
HEX_2ASC: PUSH ACC
    LCALL HEX_TO_ASC
    MOV  B,A      ; B = Lo byte
    POP  ACC
    SWAP A
    LCALL HEX_TO_ASC
    RET

;*****
;* Convert hex (0-F) to ASCII *
;* Input  : ACC (HEX code) *
;* Output : ACC (ASCII code) *
;*****
;
HEX_TO_ASC: PUSH PSW
    CLR  CY
    ANL  A,#00001111B
    CJNE A,#09H,HTOA1
    SETB CY
HTOA1:  JNB  CY,HTOA2    ;> 0AH

```

```

    ORL A,#30H    ;< 0AH
    SJMP END_HTOA
HTOA2:  SUBB A,#09H    ;> 0AH
    ORL A,#40H
END_HTOA: POP PSW
    RET
;***** CONV 1BYTE(BIN) TO 3BYTE(ASCII) *****
;TX_3ASC: lcall step ;1byte to 3digit(0-255)
    mov a,r3
    lcall HEX_TO_ASC ;decimal value to Ascii(0-9)
    lcall TX_BYTE
    mov a,r2
    lcall HEX_TO_ASC
    lcall TX_BYTE
    mov a,r1
    lcall HEX_TO_ASC
    lcall TX_BYTE
    ret
;*****
;* Send 1 byte BCD to ASCII form *
;* Input : ACC = Binary *
;* Output : BCD To rs232 ch *
;* Register: ACC.,B. *
;*****
;
SEND_BCD: CLR TI
    MOV B,A
    SWAP A
    ACALL SBCD
    MOV A,B

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SBCD:  ANL  A,#0FH
        CJNE A,#0AH,SBCD2
SBCD2:  JC   SBCD3
        ADD  A,#55
        SJMP SBCD4
SBCD3:  ORL  A,#30H
SBCD4:  CLR  ES
        MOV  SBUF,A
        JNB  TI,$
        CLR  TI
        SETB ES
        RET
;*****
;* PRINTSTR = SEND STRING *
;* FUNTION = Send string at memory - *
;* location after instruction*
;* lcall PRINT instruction *
;* end of string must be 00 *
;*****
;
;Note that the end of string table must be 00
;NOTE : when used this sub routine must not enable ES
;
PRINTSTR: POP  DPH   ; Get Pointer(stack)
          POP  DPL
          PUSH IE
          PUSH ACC
          CLR  ES
STSD0:   CLR  A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        MOVC  A,@A+DPTR
        JZ   STSD2
STSD1:  CLR   TI
        MOV   SBUF,A
        JNB  TI,$      ;Wait to send
        CLR  TI
        INC  DPTR
        SJMP STSD0
STSD2:  POP   ACC
        POP  IE
        INC  DPTR
        PUSH DPL
        PUSH DPH
        RET
;*****
;* RX_BYTE = Receive byte      *
;* FUNCTION : Receive 1 byte from *
;*          serial port.      *
;* Output   : Register ACC.    *
;*****
;
RX_BYTE: PUSH  IE
        CLR   ES
        CLR   RI
        JNB  RI,$
RX_BT1: MOV   A,SBUF
        CLR  RI
        POP  IE
        RET

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;*****
;* TX_BYTE = 1 BYTE SEND      *
;* FUNCTION = Send data in A to serial*
;*      port.                *
;* INPUT  = Data in A to transmit *
;* OUTPUT = Serial port       *
;*****
;
TX_BYTE:  PUSH  IE
          CLR   ES
          CLR   TI
          MOV   SBUF,A
          JNB  TI,$
          CLR  TI
          POP  IE
          RET

          $INCLUDE "XDELAY.ASM"
          $INCLUDE "XRS232.ASM"
          $include "k_lcd2.asm"
          $INCLUDE "XCONV_1.ASM"
          $include "math.asm"
          $include "H.txt"
          $INCLUDE "TEST_DA.ASM"
          END

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ธีรวัฒน์ ประกอบผล. การประยุกต์และใช้งานไมโครคอนโทรลเลอร์. สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น) : กรุงเทพมหานคร, 2543

ไพรัช รัชชัยพงษ์. ทฤษฎีและการใช้งานไมโครโปรเซสเซอร์และไมโครคอมพิวเตอร์. บริษัทซีเอ็ดยูเคชั่น จำกัด (มหาชน) : กรุงเทพมหานคร, 2521.

สุนทร วิฑูสรพจน์. การใช้งานไมโครคอนโทรลเลอร์ ตระกูล 8051. บริษัทซีเอ็ดยูเคชั่น จำกัด (มหาชน) : กรุงเทพมหานคร, 2537.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อผู้ทำปฏิญานិพนธ์	นายจักรพันธ์ สมุทรไชยกิจ
วันเดือนปีเกิด	18 มกราคม 2523
สถานที่เกิด	โรงพยาบาลศิริราช
ภูมิลำเนาเดิม	จังหวัดปทุมธานี
ที่อยู่ปัจจุบัน	25/4 หมู่ 5 ตำบลหน้าไม้ อำเภอลาดหลุมแก้ว จังหวัดปทุมธานี 12140
โทรศัพท์	0-2977-6840
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนชุมชนวัดหน้าไม้
มัธยมศึกษาตอนต้น	โรงเรียนปทุมวิไล
ประกาศนียบัตรวิชาชีพ (ปวช.)	วิทยาลัยเทคนิคปทุมธานี
ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)	วิทยาลัยเทคนิคปทุมธานี
ปริญญาตรี	สาขาวิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
คติพจน์	ตนเป็นที่พึ่งแห่งตน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อผู้ทำปฏิญยานิพนธ์	นายนรินทร์ ปานทอง
วันเดือนปีเกิด	5 ตุลาคม 2522
สถานที่เกิด	โรงพยาบาลเพชรบุรี
ภูมิลำเนาเดิม	จังหวัดเพชรบุรี
ที่อยู่ปัจจุบัน	969 หมู่ 3 ตำบลชะอำ อำเภอชะอำ จังหวัดเพชรบุรี
โทรศัพท์	0-3245-1368
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนเทศบาล 2 วัดไทรย้อย
มัธยมศึกษาตอนต้น	โรงเรียนเทศบาล 2 วัดไทรย้อย
ประกาศนียบัตรวิชาชีพ (ปวช.)	วิทยาลัยเทคนิคเพชรบุรี
ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)	วิทยาลัยเทคนิคเพชรบุรี
ปริญญาตรี	สาขาวิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
คติพจน์	ทำวันนี้ให้ดีที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อผู้ทำปฏิญานិพนธ์	นายวินัย กระจ่างแก้ว
วันเดือนปีเกิด	29 ธันวาคม 2522
สถานที่เกิด	โรงพยาบาลศิริราช
ภูมิลำเนาเดิม	จังหวัดปทุมธานี
ที่อยู่ปัจจุบัน	24/1 หมู่ 5 ตำบลเชียงรากน้อย อำเภอสามโคก จังหวัดปทุมธานี 12140
โทรศัพท์	0-02977-6231
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนวัดสองพี่น้อง
มัธยมศึกษาตอนต้น	โรงเรียนสามโคก
ประกาศนียบัตรวิชาชีพ (ปวช.)	เทคโนโลยีแหลมทอง
ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)	วิทยาลัยเทคนิคปทุมธานี
ปริญญาตรี	สาขาวิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
คติพจน์	ความรู้ที่ได้มา ไม่ได้เกิดขึ้นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อผู้ทำปฏิญานินพนธ์	นายสมพงษ์ วิจิตรโสภณ
วันเดือนปีเกิด	18 กุมภาพันธ์ 2517
สถานที่เกิด	โรงพยาบาลจุฬาลงกรณ์
ภูมิลำเนาเดิม	จังหวัดกรุงเทพมหานคร
ที่อยู่ปัจจุบัน	92 หมู่ 2 แขวงทับยาว เขตลาดกระบัง จังหวัดกรุงเทพมหานคร 10520
โทรศัพท์	0-1497-4058
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนมาเลียดลีย์
มัธยมศึกษาตอนต้น	โรงเรียนเทพศิรินทร์ร่วมเกล้า
ประกาศนียบัตรวิชาชีพ (ปวช.)	เทคโนโลยีกรุงเทพ
ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)	วิทยาลัยเทคนิคมีนบุรี
ปริญญาตรี	สาขาวิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
คติพจน์	ทำเพื่อแม่ และตัวของเรา ให้ดีที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้